

**VŠB – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra telekomunikační techniky**

**Alarmní a ovládací systém využívající mobilní telekomunikační  
sít'**

**Alarm and Control System Using GSM**

## Zadání diplomové práce

Student:

**Bc. Jan Hruška**

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2612T059 Mobilní technologie

Téma:

Alarmní a ovládací systém využívající mobilní telekomunikační síť.  
Alarm and Control System Using GSM.

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Navrhněte a realizujte alarmní a ovládací systém řízený soudobým mobilním telefonem.
2. Vytvořte programové vybavení řídicí jednotku systému, použijte jednodeskový počítač s procesorem architektury ARM.
3. Vytvořte aplikaci pro mobilní telefon pod OS Android.
4. Vytvořte kompletní technickou dokumentaci systému.

Seznam doporučené odborné literatury:

1. Krejčířík, A. : SMS - střežení a ovládání objektů pomocí mobilu a SMS. BEN, 2004, ISBN 80-7300-082-2.
2. Uhlář, J. : Technická ochrana objektů II - Elektrické zabezpečovací systémy. Policejní akademie České republiky v Praze, 2009. ISBN 978-80-7251-313-0.
3. Fennelly, L.J. : Effective Physical Security. Elsevier, 2013. ISBN 978-0-12-415892-4.
4. Baker, P.R. - Benny, D.J. : The Complete Guide to Physical Security. CRC Press, Boca Raton, 2013. ISBN 978-1-4200-9963-8.


Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

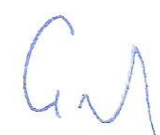
Vedoucí diplomové práce: **Ing. Radek Novák, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 29.04.2016




  
doc. Ing. Miroslav Vozňák, Ph.D.  
vedoucí katedry

  
prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## Prohlášení studenta

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: 1. července 2016

..........  
podpis studenta

## **Poděkování**

Rád bych poděkoval Ing. Radku Novákovi, Ph.D. za odbornou pomoc a konzultaci při vytváření této diplomové práce.

## **Abstrakt**

Cílem této diplomové práce je návrh a realizace alarmního a ovládacího systému řízeného mobilním telefonem.

Obsahem práce je popis použitých open source nástrojů a dalších komponent systému použitých ve funkci řídicí jednotky a senzorového uzlu. Dále je popsána realizace zapojení jednotlivých částí systému a jejich napájení.

V další části se věnuje instalaci a konfiguraci frameworku ago control a kamerového systému ZoneMinder, nastavení zabezpečeného připojení a jejich propojení. Je zde uvedena implementace programu na deskách Arduino a modulů frameworku ago control pro komunikaci s Arduino deskou v roli brány a GSM modemem. Pro mobilní telefony s OS Android byla také vytvořena aplikace pro ovládání systému.

## **Klíčová slova**

Arduino, ARM, ago control, alarmní systém, SMS, ZoneMinder, Android

## **Abstract**

Goal of this thesis is design and implementation of alarm and control system managed by mobile phone.

Content of this thesis includes description of open source tools and other components of the system used as control unit and sensor node. The thesis also includes wiring of individual parts of the system and its power supply.

Next part of the thesis is focused on installation and configuration of the framework ago control and video surveillance system ZoneMinder, configuration of secure access and its connection. It also includes implementation of application for Arduino boards and ago control framework modules used for communication with Arduino board serving as gateway and GSM modem. Application for mobile phones with Android OS was also created for managing the system.

## **Key words**

Arduino, ARM, ago control, alarm system, SMS, ZoneMinder, Android

# Seznam použitých symbolů

Symbol	Jednotky	Význam symbolu
I	A	Proud
R	$\Omega$	Odpor
U	V	Napětí
c	m/s	Rychlost světla ve vakuu
f	Hz	Frekvence
$\lambda$	m	Vlnová délka

## Seznam použitých zkratek

Zkratka	Význam
<b>ACK</b>	Acknowledgment
<b>A/D</b>	Analog-to-Digital
<b>AES</b>	Advanced Encryption Standard
<b>AMQP</b>	Advanced Message Queuing Protocol
<b>API</b>	Application Programming Interface
<b>AT</b>	Attention
<b>DNR</b>	Digital Noise Reduction
<b>EEPROM</b>	Electrically Erasable Programmable Read-Only Memory
<b>FIFO</b>	First In, First Out
<b>FTP</b>	File Transfer Protocol
<b>GPG</b>	GNU Privacy Guard
<b>GPRS</b>	General Packet Radio Service
<b>GPS</b>	Global Positioning System
<b>GSM</b>	Global System for Mobile Communications
<b>HDMI</b>	High-Definition Multi-media Interface
<b>HW</b>	Hardware
<b>I<sup>2</sup>C</b>	Inter-Integrated Circuit
<b>IDE</b>	Integrated Development Environment
<b>IP</b>	Internet Protocol
<b>IR</b>	Infrared
<b>JSON</b>	JavaScript Object Notation
<b>LED</b>	Light-Emitting Diode
<b>MISO</b>	Master In, Slave Out
<b>MOSI</b>	Master Out, Slave In
<b>OS</b>	Operating System
<b>OTG</b>	On-The-Go
<b>PHP</b>	Hypertext Preprocessor
<b>PIR</b>	Passive Infrared
<b>PTZ</b>	pan-tilt-zoom



---

<b>PWM</b>	Pulse Width Modulation
<b>QVGA</b>	Quarter Video Graphics Array
<b>RAM</b>	Random Access Memory
<b>RFID</b>	Radio Frequency Identification
<b>RISC</b>	Reduced Instruction Set Computing
<b>RPC</b>	Remote Procedure Call
<b>SCK</b>	Serial Clock
<b>SD</b>	Secure Digital
<b>SFTP</b>	SSH File Transfer Protocol
<b>SIM</b>	Subscriber Identity Module
<b>SMS</b>	Short Message Service
<b>SQL</b>	Structured Query Language
<b>SPI</b>	Serial Peripheral Interface
<b>SRAM</b>	Static Random Access Memory
<b>SS</b>	Slave Select
<b>SSL</b>	Secure Sockets Layer
<b>SW</b>	Software
<b>UART</b>	Universal Asynchronous Receiver and Transmitter
<b>UPS</b>	Uninterruptible Power Supply
<b>USB</b>	Universal Serial Bus
<b>WDR</b>	Wide Dynamic Range
<b>fps</b>	frames per second

---

## Seznam ilustrací

Číslo ilustrace	Název ilustrace	Číslo stránky
1.1	IP kamera Hikvision DS-2CD2032-I	18
1.2	IDE Arduino Software	20
1.3	Jednodeskový počítač pcDuino3 Nano	21
1.4	Logo frameworku ago control	22
1.5	GSM modem Huawei E1750	23
1.6	Schéma zapojení bezdrátového spínače RSL	24
1.7	Blokové schéma navrženého systému	25
2.1	Arduino Pro Mini	26
2.2	Bezdrátový modul HOPERF RFM69W	27
2.3	Schéma zapojení Arduino Pro Mini	28
2.4	Schéma zapojení bezdrátového modulu RFM69W	28
2.5	Bezdrátový modul HOPERF RFM85	29
2.6	Schéma zapojení konektoru pro bezdrátový modul RFM85	29
2.7	Magnetický kontakt MC-38	29
2.8	Schéma zapojení konektoru pro magnetický kontakt	30
2.9	Modul PIR čidla SB00322A-1	30
2.10	Schéma zapojení konektoru pro PIR čidlo SB00322A-1	31
2.11	Senzor teploty DS18B20 (pouzdro TO-92)	31
2.12	Schéma zapojení senzoru teploty DS18B20	32
2.13	Schéma zapojení senzoru teploty a vlhkosti DHT11	32
2.14	Schéma zapojení fotorezistoru	33
2.15	Schéma zapojení stabilizátoru napětí MCP1702	35
2.16	Schéma zapojení napěťového děliče	36
2.17	Schéma zapojení signalizace síťového zdroje Mean Well	37
3.1	Hlavní stránka ZoneMinderu po instalaci	40
3.2	Stránka ZoneMinderu pro přidání nového monitoru	41
3.3	Zobrazení zařízení v Android aplikaci	57

## Seznam tabulek

Číslo tabulky	Název tabulky	Číslo stránky
1.1	Porovnání vybraných modelů Arduino desek	19
2.1	Vliv nastavení rozlišení teploty na dobu převodu teploty	31
2.2	Odběr proudu modulu RFM69W	33
2.3	Naměřený odběr proudu Arduino Pro Mini ve vybraných provozních režimech	34
2.4	Naměřený odběr proudu Arduino Pro Mini po odstranění LED a nap. stabilizátoru	34
2.5	Technické parametry zdroje Mean Well PSC-100A-C	37
2.6	Technické parametry DC/DC měniče Mean Well SD-15A-5	37
2.7	Technické parametry akumulátoru GLP 26-12	38
2.8	Naměřený odběr proudu IP kamery a řídicí jednotky	38
3.1	Příkazy pro ovládání zařízení frameworku ago control	52
3.2	Události pro komunikaci s frameworkem ago control	52
3.3	Typy zařízení implementované v ago control klientovi	52
3.4	Příkazy použité pro komunikaci s frameworkem ago control	56

# Obsah

Úvod.....	- 15 -
1 Návrh alarmního a ovládacího systému.....	- 16 -
1.1 Senzory alarmního systému .....	- 16 -
1.1.1 Magnetický kontakt.....	- 16 -
1.1.2 Pasivní infračervené čidlo.....	- 16 -
1.2 Kamerový systém.....	- 17 -
1.2.1 ZoneMinder.....	- 17 -
1.2.2 IP kamera.....	- 17 -
1.3 Arduino platforma .....	- 18 -
1.3.1 Hardware.....	- 18 -
1.3.2 Software.....	- 19 -
1.4 Řídicí jednotka systému.....	- 20 -
1.4.1 pcDuino3 Nano .....	- 21 -
1.4.2 Operační systém.....	- 22 -
1.4.3 Úložiště.....	- 22 -
1.4.4 ago control .....	- 22 -
1.4.5 GSM modem Huawei E1750 .....	- 23 -
1.5 Conrad RSL.....	- 23 -
1.6 Funkce systému .....	- 24 -
2 Zapojení systému .....	- 26 -
2.1.1 Arduino Pro Mini .....	- 26 -
2.1.2 HOPERF RFM69W.....	- 27 -
2.2 Zapojení senzorového uzlu.....	- 27 -
2.2.1 Arduino Pro Mini .....	- 27 -
2.2.2 HOPERF RFM69W.....	- 28 -
2.2.3 HOPERF RFM85 .....	- 29 -
2.2.4 Magnetický kontakt.....	- 29 -
2.2.5 Pasivní infračervené čidlo.....	- 30 -
2.2.6 Senzor teploty DS18B20 .....	- 31 -
2.2.7 Senzor teploty a vlhkosti DHT11 .....	- 32 -
2.2.8 Fotorezistor .....	- 33 -

---

2.3	Napájení senzorového uzlu .....	- 33 -
2.3.1	Baterie.....	- 35 -
2.3.2	Měření napětí baterie.....	- 35 -
2.3.3	Spotřeba senzorového uzlu .....	- 36 -
2.4	Zapojení řídicí jednotky systému .....	- 36 -
2.5	Napájení řídicí jednotky systému a IP kamery .....	- 36 -
2.5.1	Zálohovací akumulátor .....	- 37 -
2.5.2	Spotřeba .....	- 38 -
3	SW implementace a nastavení .....	- 39 -
3.1	Operační systém jednodeskového počítače.....	- 39 -
3.2	Instalace kamerového systému ZoneMinder .....	- 39 -
3.3	Konfigurace kamerového systému ZoneMinder .....	- 40 -
3.4	Instalace frameworku ago control .....	- 42 -
3.5	Konfigurace zabezpečeného přístupu k frameworku ago control.....	- 43 -
3.6	Propojení kamerového systému ZoneMinder s ago control .....	- 44 -
3.7	Nastavení GSM modemu .....	- 45 -
3.8	Program senzorového uzlu .....	- 46 -
3.8.1	Inicializace a nastavení .....	- 46 -
3.8.2	Hlavní část programu.....	- 46 -
3.8.3	Obsluha přerušení.....	- 47 -
3.8.4	Zprávy zasílané ze senzorového uzlu na Arduino bránu .....	- 47 -
3.9	Program Arduino brány .....	- 48 -
3.9.1	Inicializace a nastavení .....	- 49 -
3.9.2	Hlavní část programu.....	- 49 -
3.9.3	Zprávy zasílané z Arduino brány na senzorový uzel.....	- 50 -
3.9.4	Zprávy zasílané z Arduino brány ago control klientovi.....	- 50 -
3.9.5	Zprávy zasílané ago control klientem na Arduino bránu .....	- 51 -
3.10	Modul ago control klient.....	- 51 -
3.10.1	Obsluha příkazů z frameworku ago control .....	- 52 -
3.10.2	Příjem zpráv z Arduino brány .....	- 52 -
3.10.3	Nastavení systémové služby .....	- 53 -
3.11	Modul pro komunikaci s GSM modemem.....	- 53 -
3.11.1	Knihovna Gammu .....	- 53 -

---

3.11.2 Konfigurační soubor .....	- 54 -
3.11.3 Funkce modulu .....	- 55 -
3.12 Mobilní aplikace pro OS Android .....	- 55 -
3.12.1 Funkce aplikace.....	- 56 -
Závěr .....	- 58 -
Použitá literatura.....	- 59 -
Seznam příloh.....	- 62 -

## Úvod

Alarmní a ovládací systémy v dnešní době nabízí mnoho výrobců v různých cenových relacích. V závislosti na požadavcích na systém kladených je možný výběr od základních systémů s několika senzory, omezenými možnostmi dalšího rozšíření a vzdálené správy až po komplexní systémy, které spojují zabezpečení a ovládání objektu s možností autonomního řízení chodu objektu.

Častým omezením komerčních řešení je provázanost s produkty daného výrobce, již při pořizování základních prvků systému je tedy vhodné zvážit možnosti budoucího rozšíření systému. Možnost využití dostupných zařízení pro zabezpečení a ovládání objektu bez nutnosti výběru produktů konkrétního výrobce nabízejí open source systémy pro domácí automatizaci.

Jednotlivé open source systémy mohou podporovat produkty různých výrobců, díky otevřenosti zdrojového kódu je však možné řadu výrobců podporovaných zařízení dále rozšiřovat, případně k systému připojit vlastní navržená zařízení.

Tato práce se zabývá návrhem a realizací alarmního a ovládacího systému, který je možné ovládat pomocí mobilního telefonu. Navržený systém je založen na jednodeskovém počítači a open source frameworku ago control v roli řídicí jednotky systému, která je propojena s dalšími zařízeními na platformě Arduino. Navržený systém umožňuje propojení s kamerovým systémem a je možné jej ovládat webovým prohlížečem, aplikací pro mobilní telefon s OS Android nebo SMS příkazy z jakéhokoliv mobilního telefonu.

V úvodní části práce je popsán návrh systému a jeho hlavních HW a SW prvků. Dále je uvedena realizace zapojení HW části systému na platformě Arduino s popisem jednotlivých komponent a ve 3. kapitole pak implementace a konfigurace jednotlivých SW komponent systému.

# 1 Návrh alarmního a ovládacího systému

Alarmní systémy slouží pro detekci narušení hlídaného objektu a jsou běžně používány pro střežení soukromých i komerčních objektů. Součástí alarmních systémů jsou zejména senzory pro samotnou detekci narušení prostoru, ústředny pro vyhodnocování stavů připojených senzorů a zařízení pro indikaci narušení objektu. Konkrétní složení systému je závislé na typu a velikosti střeženého objektu a požadavcích na něj kladených.

Alarmní systémy lze podle typu komunikace mezi jednotlivými jeho prvky rozdělit na drátové a bezdrátové. Výhodou bezdrátových systémů je rychlost a jednoduchost jejich instalace s minimálními stavebními zásahy, jelikož nevyžadují zabudování vodičů v objektu. Nevýhodou je naopak napájení bateriemi a podle typu a velikosti střeženého objektu také možný omezený dosah.

Pro výše uvedené výhody je dále alarmní systém navrhován jako bezdrátový. V následujících kapitolách je uveden základní popis použitých senzorů alarmního systému a dalších HW a SW komponent a nástrojů použitých při realizaci systému.

## 1.1 Senzory alarmního systému

### 1.1.1 Magnetický kontakt

Magnetické kontakty lze použít pro detekci otevření dveří, oken nebo bran. Skládají se z jazýčkového kontaktu a permanentního magnetu. Jazýčkový kontakt je mechanický spínač ovládaný magnetickým polem, který tvoří hermeticky uzavřená skleněná trubička, v níž jsou umístěny dva feromagnetické jazýčky.

V klidovém stavu je jazýčkový kontakt sepnut magnetickým polem permanentního magnetu. Při oddálení magnetu dojde k rozepnutí kontaktu a vyvolání poplachu.

Při montáži se jazýčkový kontakt umísťuje na rám oken či dveří, magnet poté na samotné křídlo okna či dveře. Magnetické kontakty jsou k dostání v provedení pro povrchovou nebo skrytou montáž.

### 1.1.2 Pasivní infračervené čidlo

PIR čidla jsou jedním z nejrozšířenějších druhů čidel používaných v alarmních systémech a v systémech pro automatické osvětlení. Pracují na principu zachycení změn vyzařování elektromagnetického záření v infračerveném pásmu kmotočtového spektra. Jsou tedy určeny k zachycení pohybu objektů s teplotou odlišnou od okolí.

Umístění PIR čidla by mělo být takové, aby nebylo vystaveno vlivům ventilace, vyzařování světla (slunce, reflektory automobilu) a jiným proměnlivým zdrojům tepla (např. plynové a podlahové topení, komín). Měla by být umístěna na pevném podkladu, a jelikož se jedná o pasivní čidla nevyzařující energii, je možné do prostoru instalovat více PIR čidel, aniž by docházelo k vzájemnému ovlivnění.

Zorné pole PIR čidla je rozděleno na aktivní a neaktivní zóny nejčastěji Fresnelovou čočkou, přičemž u některých čidel je možné změnou čočky docílit změny detekční charakteristiky.



Další informace o zabezpečovacích systémech lze nalézt v [1], [2], [3] případně [4].

## 1.2 Kamerový systém

Kamerový systém lze použít jak samostatně, tak jako součást zabezpečovacího systému. Umožňuje detekci pohybu, záznam narušení sledovaného objektu a případně identifikaci narušitele. Kamerové systémy jsou také používány pro prevenci kriminality v ulicích měst, budovách či na silnicích.

Moderní kamerové systémy mohou být vybaveny přísvitem pomocí IR diod pro osvětlení scény za tmy a šera, přiblížením obrazu, sledováním pohybu objektu nebo detekcí obličejů.

### 1.2.1 ZoneMinder

ZoneMinder je open-source kamerový zabezpečovací systém vyvinutý pro použití s OS Linux. Podporuje připojení analogových i digitálních kamer, které mohou být připojeny přes USB i síťové rozhraní. Základními funkcemi ZoneMinderu jsou analýza obrazu včetně detekce pohybu a nahrávání záznamu. ZoneMinder je možné použít pro zabezpečení domácností i firem, přičemž poskytuje možnost horizontálního škálování a vytvoření clusteru ZoneMinder serverů.

ZoneMinder je možné provozovat na kterékoliv distribuci Linuxu podporující Video4Linux a vyžaduje webový server Apache, databázi MySQL a podporu programovacího jazyku Perl. Je naprogramován převážně v jazycích C++, Perl a PHP.

ZoneMinder disponuje webovým rozhraním umožňujícím podrobné nastavení jednotlivých kamer, sledování aktuálního obrazu z kamer i přehrávání záznamů. U PTZ kamer lze také ovládat jejich pohyb a přiblížení obrazu. Každé kaměře je možné pomocí zón nastavit oblasti, ve kterých má být detekován pohyb (včetně citlivosti), případně oblasti, které mají ignorovány. Detekci pohybu analýzou obrazu lze také nahradit spouštěním událostí externími aplikacemi nebo zařízeními.

Zobrazení náhledů záznamu, kvalitu, formát i velikost přehrávaného záznamu lze ovlivnit v závislosti na rychlosti sítě integrovanými profily. Pro přístup k webovému rozhraní je možné využít zabezpečené připojení, případně také uživatelské účty s podrobně specifikovaným oprávněním.

Automatickou archivaci nebo naopak smazání starších záznamů je možné nadefinovat ve filtrech. Záznamy z kamer je možné automaticky nahrávat na FTP nebo SFTP server a s každou vytvořenou událostí je možné zasílat upozornění formou e-mailové nebo SMS zprávy.

Další informace o systému ZoneMinder lze nalézt na [9].

### 1.2.2 IP kamera

Digitální IP kamery oproti analogovým obvykle vynikají vysokým rozlišením a kvalitou obrazu. Ke své funkci nevyžadují připojení ke kamerovému rekordéru a je možné je jednoduše připojit k počítačové síti. Nastavení parametrů IP kamery je obvykle možné provést ve webovém rozhraní.

V této práci byla použita IP kamera Hikvision DS-2CD2032-I určená pro použití ve venkovním prostředí, která disponuje rozlišením obrazu až 3 Mpx při 20 snímcích za vteřinu a IR LED pro noční přísvícení s dosahem až 30 m.

U kamery je možné využít dual stream přenosu, kdy rozlišení obrazu 2. streamu je možné nastavit už od QVGA. Kamera obsahuje funkce detekce pohybu, křížení plochy, detekce obličejů, DNR pro snížení šumu a WDR pro zvýšení kontrastu obrazu. Umožňuje nastavení omezení datového toku užitečné např. pro sledování obrazu na mobilním zařízení s nízkou rychlostí připojení. Připojení k počítačové síti zajišťuje konektor RJ-45, kterým může být kamera s využitím PoE 802.3af také napájena. Pro napájení kamery je možné použít také souosý napájecí konektor. Spotřeba kamery dosahuje maximálně 5 W bez IR přísvisitu a 7 W s IR přísvisícením. [11]



Obrázek 1.1: *IP kamera Hikvision DS-2CD2032-I [11]*

### 1.3 Arduino platforma

Arduino je open-source platforma navržená s ohledem na jednoduchost použití jak software, tak hardware. Tato platforma vznikla v roce 2005 na Interaction Design Institute Ivrea v Itálii a je založena na open-source frameworku Wiring a mikrokontrolérech ATmega od firmy Atmel. Původním cílem bylo vytvoření levné a jednoduché prototypovací platformy pro studenty, která umožní rychlý vývoj nových zařízení.

#### 1.3.1 Hardware

Základem většiny dostupných modelů Arduina je mikrokontrolér ATmega z rodiny AVR, kterou vyvinula firma Atmel v roce 1996 v Norsku. ATmega jsou osmibitové mikrokontroléry typu RISC s Harvardskou architekturou, a tedy oddělenou pamětí pro data a program. Mikrokontroléry z rodiny AVR byly jedněmi z prvních, které pro uložení programu použily paměť typu flash.

Arduino desky nejčastěji používají mikrokontroléry ATmega328P, ATmega32U4 nebo ATmega2560, dříve pak také ATmega8, ATmega168 a ATmega1280. Jednotlivé mikrokontroléry se liší zejména počtem vstupně výstupních pinů, velikostí pamětí RAM a flash. Většina modelů Arduino desek kromě samotného mikrokontroléru obsahuje krystalový oscilátor, stabilizátor napětí, konektory analogových a digitálních vstupně výstupních pinů, tlačítko pro reset mikrokontroléru a několik LED pro indikaci provozu a sériové komunikace. Některé modely mají také napájecí a USB konektor, který umožňuje připojení k počítači, programování mikrokontroléru a sériovou komunikaci přes virtuální sériový port. Vytvoření portu a samotnou komunikaci zajišťuje k tomu účelu naprogramovaný mikrokontrolér ATmega16U2, na dřívějších modelech byl použit mikrokontrolér ATmega8U2, případně čip firmy FTDI. Porovnání parametrů vybraných Arduino modelů je uvedeno v tabulce 1.1.

Tabulka 1.1: *Porovnání vybraných modelů Arduino desek*

označení desky	Arduino Uno	Arduino Pro Mini	Arduino Mega
mikrokontrolér	ATmega328P	ATmega328P	ATmega2560
provozní napětí	5 V	3,3 V nebo 5 V (podle modelu)	5 V
počet digitálních pinů	14	14	54
počet digitálních PWM pinů	6	6	15
počet analogových pinů	6	6	16
flash paměť	32 kB (0,5 kB bootloader)	32 kB (0.5 kB (bootloader)	256 kB (8 kB bootloader)
SRAM	2 kB	2 kB	8 kB
EEPROM	1 kB	1 kB	4 kB
taktovací frekvence	16 MHz	8 MHz (3,3 V model) 16 MHz (5 V model)	16 MHz
délka	68,6 mm	33 mm	101,52 mm
šířka	53,4 mm	18 mm	53,3 mm

Mikrokontroléry v Arduino deskách mají v paměti předprogramovaný bootloader, který jejich programování zjednodušuje, jelikož není potřeba použít externí hardwarový programátor. Bootloader v modelu Arduino Uno má velikost 0,5 kB, v případě potřeby využití celé flash paměti mikrokontroléru je možné bootloader vymazat.

K deskám Arduino Uno či Mega je možné připojit přídatné moduly, které jsou označovány jako „shields“. Tyto moduly mohou standardním deskám přidávat další funkcionalitu jako připojení k síťovému rozhraní (Ethernet Shield), ovládání stejnosměrných motorů (Motor Shield), připojení k síti GSM (GSM/GPRS shield) či získání dat o poloze (GPS Shield).

Vzhledem k otevřenosti platformy a dostupnosti dokumentace všech modelů hardware mohlo vzniknout mnoho jejich klonů (např. Freeduino, Boarduino, Seeeduino). Tyto klony mohou mít kompatibilní formát s některým z originálních Arduino modelů, nebo mohou být kompatibilní jen po stránce software. Klony mohou také možnosti původního Arduino modelu rozšiřovat (a obsahovat např. Wi-Fi či Bluetooth adaptér a různé senzory), nebo být naopak zaměřeny na jednoduchý návrh a co nejmenší velikost.

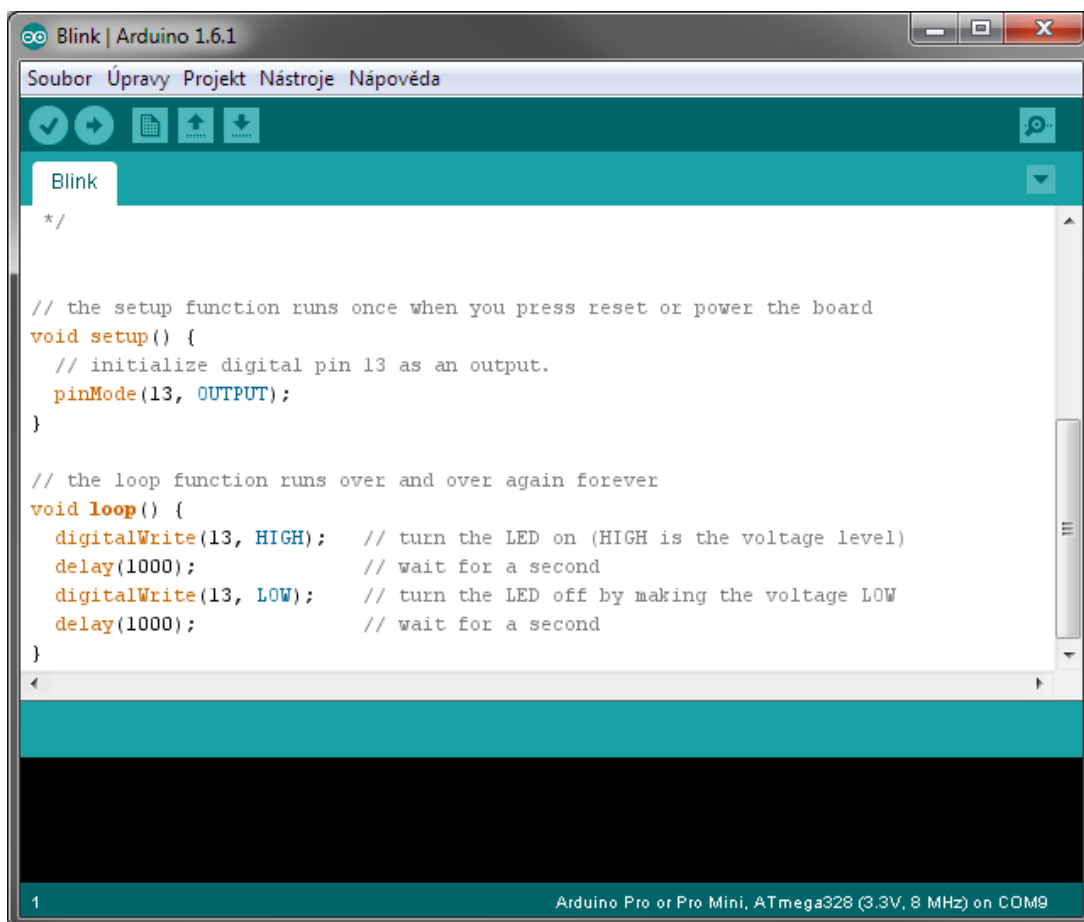
### 1.3.2 Software

Programování pro Arduino je možné ve stejnojmenném programovacím jazyce, který vychází z frameworku Wiring. Tento framework vznikl jako zjednodušení programovacího jazyku C jeho rozšířením o Wiring knihovny.

Pro psaní programů (nazvaných „sketch“) a jejich nahrání na mikrokontrolér v některém z Arduino modelů autoři Arduina vyvinuli multiplatformní vývojové prostředí (IDE) Arduino Software. Toto IDE je napsáno v programovacím jazyce Java a je založeno na open-source IDE Processing. Rozhraní Arduino IDE obsahuje editor pro psaní kódu, textovou konzoli pro výpis chybových zpráv a panel nástrojů s tlačítky pro zkompilování programu a jeho nahrání do

mikrokontroléru. Součástí IDE je také konzole pro sériovou komunikaci a ve vývojové verzi také manažér knihoven a sériový plotter, který umožňuje grafické zobrazení dat.

Arduino programovací jazyk je možné rozšiřovat pomocí knihoven, přičemž v samotném Arduino Software jsou obsaženy knihovny včetně ukázek jejich použití např. pro sériovou komunikaci, zobrazování dat na displeji, práci s pamětí nebo s ethernetovým a Wi-Fi síťovým rozhraním. Mnoho dalších knihoven pak vytváří Arduino komunita a prodejci či výrobci vybraných zařízení.



Obrázek 1.2: *IDE Arduino Software*

Další informace a dokumentaci Arduino platformy lze nalézt na [8].

### 1.4 Řídicí jednotka systému

Řídicí jednotka je hlavní část systému, která obstarává komunikaci mezi jednotlivými částmi systému. Umožňuje uchovávání dat z připojených senzorů, ovládání připojených zařízení a definování akcí pro nastalé události systému.

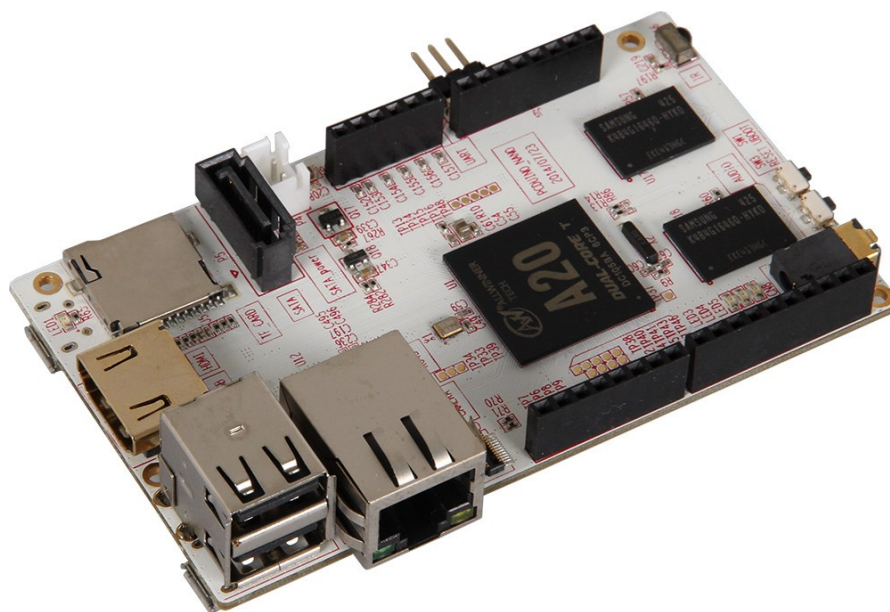
Řídicí jednotku navrženého systému tvoří jednodeskový počítač pcDuino3 Nano s GSM modemem, pevným diskem a deskou Arduino s bezdrátovým modulem. Důležitou součástí je také operační systém Linux, kamerový systém ZoneMinder a framework ago control.

### 1.4.1 pcDuino3 Nano

Jednodeskový počítač pcDuino3 Nano byl představen v roce 2014 firmou LinkSprite a je inovovanou a menší verzí jednodeskového počítače pcDuino3. Tyto počítače byly navrženy s cílem usnadnění vývoje hardwaru a softwaru jak pro výrobce, tak pro studenty a technologické nadšence. Velikost pcDuino3 Nano je 9,2 cm na délku a 5,4 cm na šířku a od ostatních jednodeskových počítačů, např. známějších modelů Raspberry Pi, se odlišuje rozložením rozšiřujících pinů shodným s rozložením pinů na desce Arduino Uno. Umožňuje tedy připojení přídavných modulů Arduino Shields. Rozšiřující rozhraní počítače zahrnuje 14 digitálních vstupně výstupních pinů (ze kterých 6 poskytuje PWM výstup) a 6 analogových vstupních pinů. Stejně jako deska Arduino Uno obsahují jednodeskové počítače pcDuino3 rozhraní SPI, I<sup>2</sup>C a UART.

Obě verze pcDuino3 používají 32bitový procesor Allwinner A20 se dvěma jádry architektury ARM Cortex-A7, které jsou taktovány na frekvenci 1 GHz. K dispozici je 1 GB operační paměti a 4 GB paměti flash. Velikost úložiště je možné navýšit vložením microSD paměťové karty do velikosti 32 GB, na které může být také nainstalovaný systém, nezávisle na interní flash paměti. Pro pcDuino3 Nano výrobce podporuje instalaci operačních systémů Linux – distribuce Ubuntu (konkrétně jeho odnož Linaro, jež vyvíjí stejnojmenná organizace) a Android.

Výhodou jednodeskového počítače pcDuino3 Nano je implementace nativní SATA sběrnice umožňující připojení pevných disků. Na rozdíl od větší klasické verze pcDuino3, verze Nano neobsahuje Wi-Fi adaptér, naopak však přidává port USB rozhraní verze 2.0 (celkem má deska tyto porty dva a jeden USB OTG pro využití s nainstalovaným operačním systémem Android). Připojení k ethernetové síti zprostředkuje gigabitové síťové rozhraní, port HDMI verze 1.4 pak poskytuje případný obrazový a zvukový výstup. Napájení desky je možné zdrojem stejnosměrného napětí 5 V připojeném k napájecímu portu microUSB. [6]



Obrázek 1.3: Jednodeskový počítač pcDuino3 Nano [6]

### 1.4.2 Operační systém

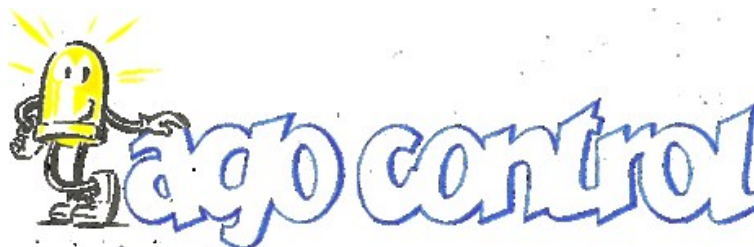
Pro potřeby této práce byl zvolen OS Linux – distribuce armbian, která je založena na aktuální 8. stabilní verzi distribuce Debian. Jelikož není potřeba grafického výstupu na samotném jednodeskovém počítači, byla zvolena varianta Jessie server bez desktopového prostředí s Linuxovým jádrem verze 3.4.105. [7]

### 1.4.3 Úložiště

Jako úložiště dat pro kamerový systém ZoneMinder je použit pevný disk Western Digital WD7500BPVT o kapacitě 750 GB a formátu 2,5“. Průměrná spotřeba disku při čtení/zápisu je podle specifikace výrobce 1,6 W a v klidovém režimu 0,65 W. [24]

### 1.4.4 ago control

Ago control je open-source framework s modulární architekturou určený pro ovládání zařízení. Byl navržen s cílem nabídnout řešení domácí automatizace a je určen k provozu na OS Linux včetně systémů běžících na procesorech ARM architektury.



Obrázek 1.4: Logo frameworku ago control [12]

Pro komunikaci mezi jednotlivými částmi frameworku ago control je použito protokolu AMQP s Apache Qpid™ brokerem. Mezi hlavní části frameworku patří:

- **Resolver** – obstarává registraci nových zařízení a překlad mezi interním a globálním ID jednotlivých zařízení
- **Timer** – slouží ke spouštění událostí závislých na času
- **Eventconfiguration** – slouží k provádění nastavených akcí na základě uživatelsky definovaných pravidlech
- **Scenarioconfiguration** – používá se k vytvoření „scén“, tedy skupin příkazů, které jsou jednotlivě prováděny a mohou být spouštěny uživatelem nebo událostmi
- **RPC Interface** – serverová část webového rozhraní poskytující API pro komunikaci s dalšími částmi frameworku
- **Web Admin** – webové rozhraní pro nastavení a sledování stavu zařízení, událostí a scénářů
- **Data Logger** – zaznamenávání všech událostí zachycených ze zařízení do logu
- **Blockly** – slouží pro vizuální definici akcí
- **SecuritySystem** – slouží pro nastavení zabezpečovacího systému, umožňuje nastavení stavů zabezpečení, zón a k nim přiřazených senzorů

Framework ago control podporuje připojení mnoha různých druhů zařízení, konkrétně např. telefonní ústředny Asterisk, zavlažovacího systému Rain8net, reléových desek IPX800-v3

připojitelných k ethernetové síti, zařízení využívajících 1-Wire sběrnici, webkamer, případně také zařízení pro domácí automatizaci, jež využívají protokol KNX, X10 nebo Z-Wave. Pro přidání podpory jiných zařízení jsou poskytovány knihovny pro programovací jazyky C++ a Python, případně JSON-RPC API. Další informace a návody k frameworku ago control lze nalézt na [12].

#### 1.4.5 GSM modem Huawei E1750

Huawei E1750 je USB GSM modem podporující jak datové připojení prostřednictvím technologií GPRS, EDGE nebo HSPA, tak příjem a odesílání SMS zpráv. Modem disponuje slotem pro mini SIM kartu, microSD paměťovou kartu, diodou pro indikaci aktuálního stavu modemu a USB konektorem pro připojení k počítači.

Modem podporuje operační systémy Microsoft Windows a Mac OS, je však možné jej použít i v aktuální 8. verzi OS Linux distribuce Debian. Pro komunikaci s modemem je možné využít projektu Gammu [31], který zahrnuje knihovny pro jazyky C++ a Python poskytující API pro ovládání modemu a utilitu pro příkazovou řádku

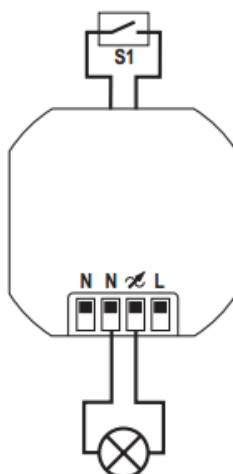


Obrázek 1.5: GSM modem Huawei E1750

#### 1.5 Conrad RSL

RSL je bezdrátový systém určený pro domácí automatizaci. Prodejcem je firma Conrad Electronic založená roku 1923 v Německu. Jednotlivé prvky systému komunikují v bezlicenčním frekvenčním pásmu ISM 433 MHz.

Systém zahrnuje např. bezdrátové dálkové ovladače, nástěnné vypínače, detektory pohybu, dveřní a okenní senzor, vestavné spínače a stmívače či ovladač rolet a žaluzií. V této práci byl pro možnost jednoduchého zapojení ve stávající elektroinstalaci použit vestavný bezdrátově ovládaný spínač RSLU určený pro instalaci pod stávající nástěnný vypínač. Spínač umožňuje spínat zařízení se spotřebou až 2 kW a schéma jeho zapojení je na obrázku 1.6. [25]



Obrázek 1.6: Schéma zapojení bezdrátového spínače RSL [25]

## 1.6 Funkce systému

Hlavní částí navrženého systému je řídicí jednotka, kterou tvoří jednodeskový počítač, GSM modem a deska Arduino Pro Mini s bezdrátovým modulem, dále označována jako Arduino brána.

Základem SW části systému je framework ago control a kamerový systém ZoneMinder. ZoneMinder umožňuje připojení IP kamery s nahráváním záznamu při detekci pohybu nebo ve spojení s ago control v závislosti na událostech systému.

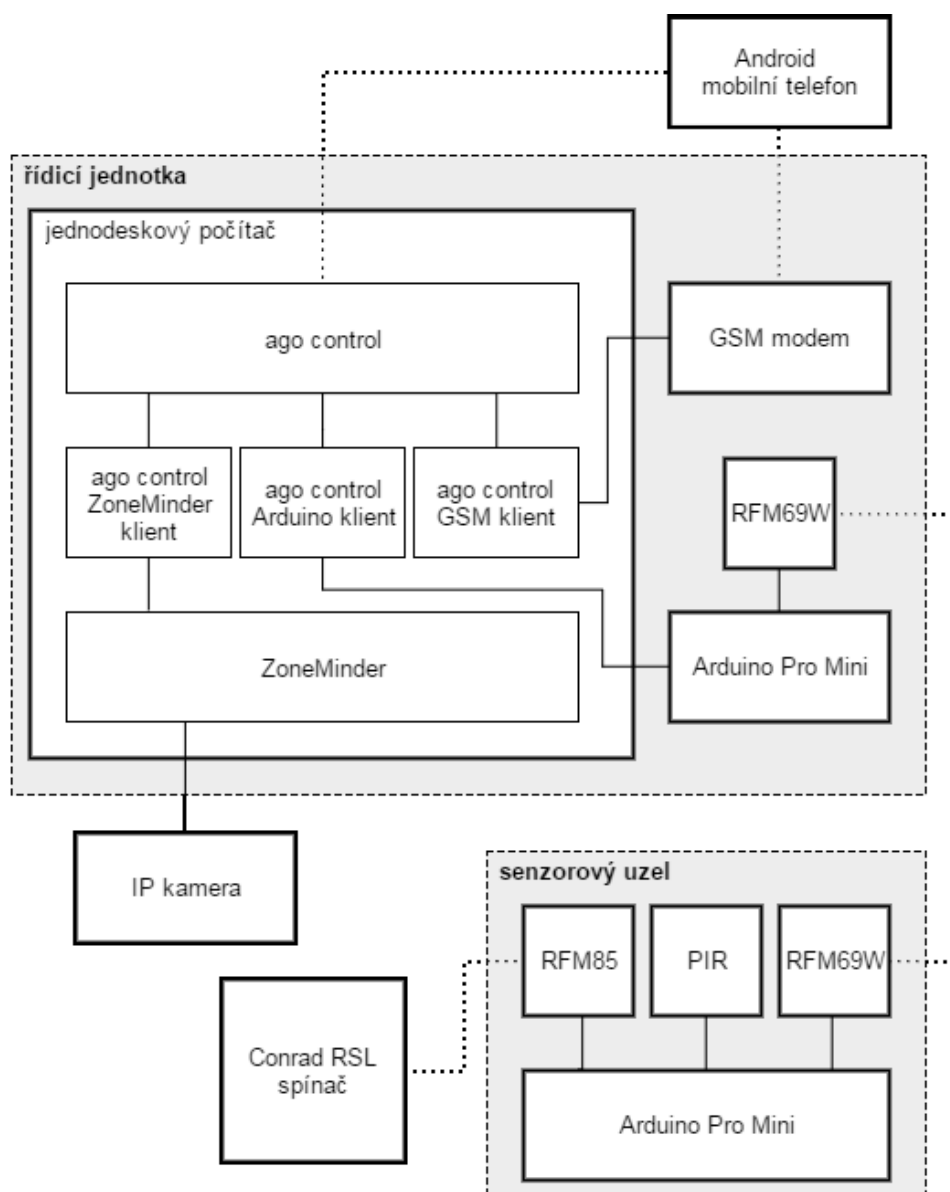
Arduino brána je k jednodeskovému počítači připojena sériovým rozhraním a je tedy možné ji případně použít s jiným jednodeskovým nebo standardním počítačem. Arduino brána slouží pro zprostředkování komunikace mezi frameworkem ago control a Arduino deskami s připojenými vstupními či výstupními zařízeními, dále označovanými jako senzorové uzly.

K senzorovým uzlům mohou být připojeny např. senzory pohybu a teploty, magnetické nebo mechanické kontakty, případně také bezdrátový modul RFM85 pro ovládání prvků systému Conrad RSL nezávisle na zvolené verzi bezdrátového modulu RFM69W.

GSM modem slouží pro ovládání systému pomocí SMS zpráv, v případě mobilního telefonu s OS Android je také možné využít vytvořené aplikace.

Blokové schéma systému je uvedeno na obrázku 1.7.





Obrázek 1.7: *Blokové schéma navrženého systému*

## 2 Zapojení systému

Obsahem této kapitoly je popis konkrétních použitých komponent a jejich zapojení při realizaci senzorového uzlu a řídicí jednotky systému. V následující části je uveden popis desky Arduino Pro Mini a bezdrátového modulu HOPERF RFM69W jež jsou použity jak v zapojení senzorového uzlu, tak v zapojení Arduino brány.

Měření uvedená v této kapitole byla provedena multimetrem UNI-T UT33C. Pro návrh schémat zapojení byl použit program EAGLE s knihovnami [38], [39] a [40].

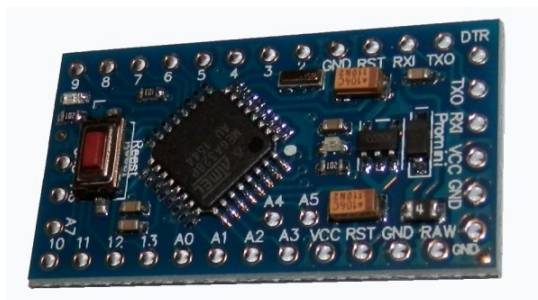
### 2.1.1 Arduino Pro Mini

Arduino Pro Mini je nejmenším modelem ze základní řady Arduino desek pro univerzální použití. Využívá stejně jako např. model Arduino Uno mikrokontrolér ATmega328P a obsahuje tedy 32 kB flash paměti pro uložení programu (z nichž je 0,5 kB využito bootloaderem), 2 kB SRAM paměti a 1 kB EEPROM paměti. Tento model je dostupný ve 2 verzích lišících se provozním napětím, které může být 3,3 V nebo 5 V a taktovací frekvencí mikrokontroléru, jež může být 8 MHz nebo 16 MHz. Tento model byl navržen a je vyráběn firmou SparkFun Electronics.

Hlavními rozdíly oproti modelu Arduino Uno je menší velikost a s ní související rozložení pinů, absence napájecího i USB konektoru a s ním také přídavného mikrokontroléru, který obstarává sériovou komunikaci s počítačem. Pro programování mikrokontroléru nebo sériovou komunikaci s počítačem je tedy potřeba použít přídavný převodník z USB na sériové rozhraní mikrokontroléru. Napájet desku je možné zdrojem stejnosměrného napětí 3,3 V nebo 5 V (v závislosti na modelu) připojeném k pinu VCC, případně napětím až 12 V připojeném k pinu RAW.

Arduino Pro Mini poskytuje 14 vstupně výstupních pinů, které obsahují volitelně využitelný vnitřní pull-up rezistor, přičemž 6 pinů (3, 5, 6, 9, 10 a 11) je možné použít jako 8bitový PWM výstup. Piny 0 a 1 slouží pro sériovou komunikaci, piny 2 a 3 je pak možné nakonfigurovat pro vyvolání přerušení mikrokontroléru. Pro SPI komunikaci slouží piny 10 (SS), 11 (MOSI), 12 (MISO) a 13 (SCK). K pinu 13 je také připojena jedna z LED integrovaných na desce.

Model Pro Mini je vybaven 8 analogovými vstupními piny (označenými A0 – A7), přičemž 6 je jich umístěno po stranách a 2 uvnitř desky. Piny A4 a A5 podporují I<sup>2</sup>C komunikaci. A/D převodník na analogových pinech poskytuje 10bitové rozlišení a rozsah měření je omezen provozním napětím v závislosti na vybrané verzi. Oproti modelu Arduino Uno nemá Arduino Pro Mini vyveden pin AREF a referenční napětí A/D převodníku je tedy možné změnit jen na interní referenční napětí mikrokontroléru ATmega328P, které je 1,1 V s tolerancí  $\pm 0,1$  V. [14] [19]

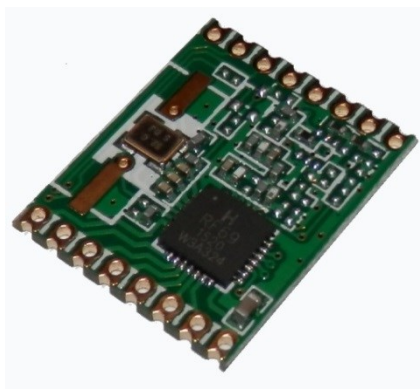


Obrázek 2.1: *Arduino Pro Mini*

### 2.1.2 HOPERF RFM69W

Bezdrátový modul RFM69W firmy HOPERF umožňuje vysílání a příjem dat ve frekvenčních pásmech 433 MHz, 868 MHz a 915 MHz. Je optimalizován pro nízkou spotřebu v režimu spánku a vysoký výstupní výkon.

Modul disponuje maximální rychlostí přenosu dat až 300 kb/s. Umožňuje využití šifry AES se 128bitovým klíčem pro šifrování dat, avšak s omezením maximální velikosti paketu na 66 B. Při příjmu i odesílání paketu je využita 66 B FIFO fronta. Chyby v přijatých paketech jsou kontrolovány cyklickým redundantním součtem. Modul v režimu spánku odebírá jen 0,1  $\mu\text{A}$ , v režimu přijímání dat odebírá 16 mA a při odesílání dat maximálně 45 mA (dostupná je však také výkonnější verze RFM69HW s odběrem až 130 mA). Modul disponuje citlivostí -120 dBm při přenosové rychlosti 1,2 kb/s a nastavitelným výstupním výkonem v rozsahu -18 až +13 dBm. Provozní napětí modulu je 1,8 až 3,6 V a pro komunikaci s mikrokontrolérem je použita SPI sběrnice. [20]



Obrázek 2.2: Bezdrátový modul HOPERF RFM69W

Pro bezdrátovou komunikaci prostřednictvím modulu HOPERF RFM69W byla použita knihovna RFM69 [21], která je určena pro použití s Arduino deskami a jejich klony využívající mikrokontroléry ATmega. Tato knihovna poskytuje API pro odesílání a příjem paketů s volitelným šifrováním, přepínání provozních režimů modulu, případně odesílání potvrzení (ACK).

V této práci jsou použity moduly pro komunikaci ve frekvenčním pásmu 433 MHz, je však také možné je nahradit moduly pro frekvenční pásmo 868 MHz.

Anténu použitého modulu tvoří vodič o délce  $\frac{1}{4}$  vlnové délky, kterou lze vypočítat podle vzorce 2.1. Délka antény tedy je 17,3 cm.

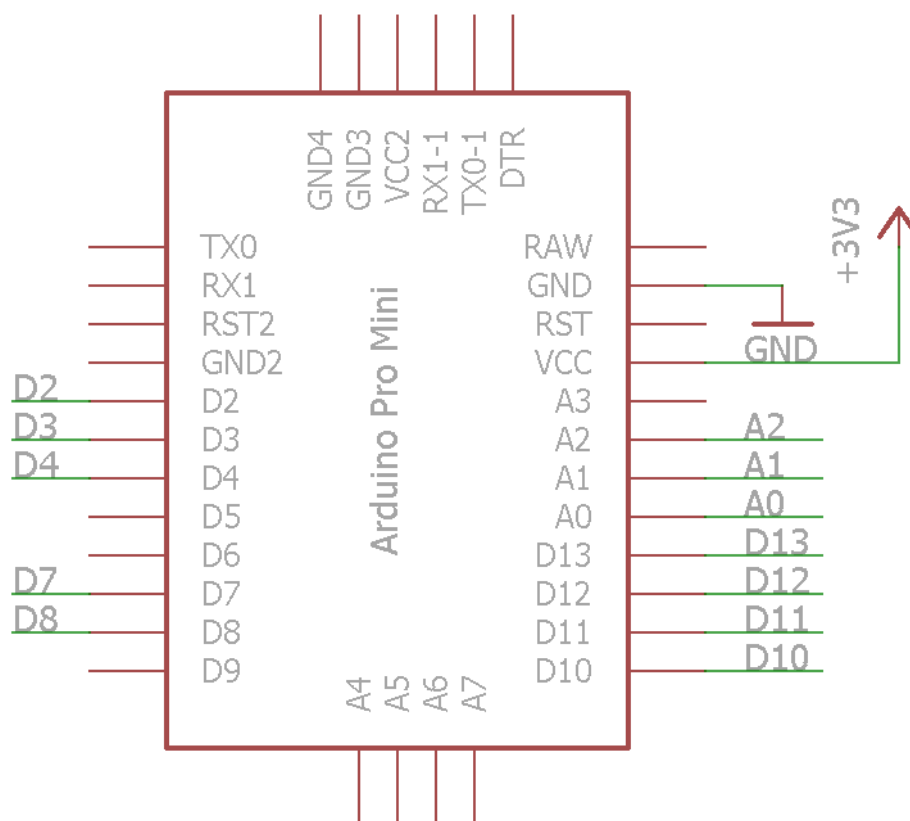
$$\lambda = \frac{c}{f} \quad (2.1)$$

$$\lambda = \frac{300 \cdot 10^6}{433 \cdot 10^6} \doteq 0,693 \text{ m} \quad (2.2)$$

## 2.2 Zapojení senzorového uzlu

### 2.2.1 Arduino Pro Mini

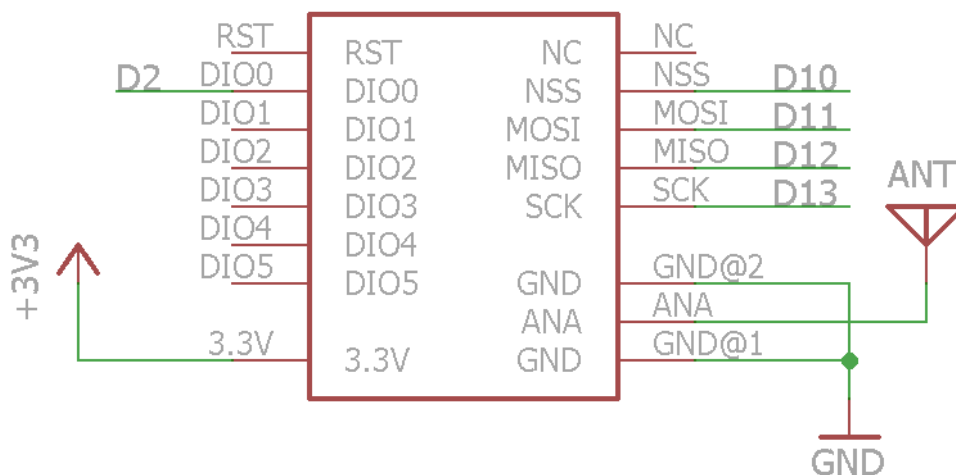
Hlavní částí senzorového uzlu je deska Arduino Pro Mini, ke které jsou připojeny použité senzory a moduly uvedené v dalších kapitolách. Arduino deska je napájena napětím 3,3 V ze stabilizátoru napětí, schéma zapojení je na obrázku 2.3.



Obrázek 2.3: Schéma zapojení Arduino Pro Mini

### 2.2.2 HOPERF RFM69W

Bezdrátový modul RFM69W je k Arduino desce připojen SPI rozhraním na pinech D10-D13. Příjem dat modulu je signalizován přerušením na pinu D2 Arduino desky. Modul je napájen stejně jako Arduino deska napěťovým stabilizátorem s výstupním napětím 3,3 V.



Obrázek 2.4: Schéma zapojení bezdrátového modulu RFM69W

### 2.2.3 HOPERF RFM85

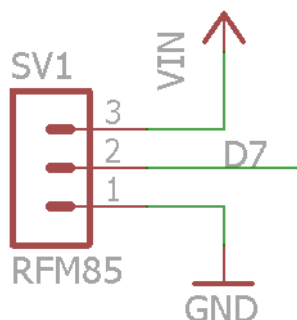
Bezdrátové moduly řady RFM85 jsou cenově dostupné ASK/OOK vysílače s vysokým výkonem pracující na frekvenci 315 MHz nebo 433 MHz. Jsou určeny pro použití v bezdrátových senzorech, ovladačích nebo zabezpečovacích systémech. Provozní napětí modulu je 2,1 až 5,5 V. Odběr proudu modulu v aktivním režimu, stejně jako jeho výstupní výkon, závisí na napájecím napětí a je 24 mA při napětí 3,3 V a 40 mA při napětí 5 V. [22]

V kombinaci s knihovnou rc-switch [23] pro Arduino desky je tento modul použit k ovládání spínačů bezdrátového systému Conrad RSL.



Obrázek 2.5: Bezdrátový modul HOPERF RFM85 [22]

Jelikož je výstupní výkon modulu závislý na napájecím napětí, není připojen k výstupnímu napětí stabilizátoru. Napájecí napětí zapojení tedy při využití tohoto modulu nesmí překročit 5,5 V. Datový pin modulu je připojen k pinu D7 na Arduino desce. Zapojení konektoru pro tento modul je provedeno podle obrázku 2.6.



Obrázek 2.6: Schéma zapojení konektoru pro bezdrátový modul RFM85

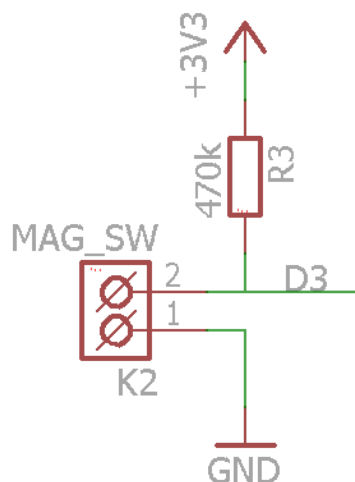
### 2.2.4 Magnetický kontakt

Použitý magnetický kontakt má označení MC-38 a je určen pro povrchovou montáž. Jednotlivé části jsou uzavřeny v plastovém pouzdru a je možné je upevnit samolepící páskou nebo šrouby. Maximální vzdálenost mezi jednotlivými částmi v klidovém stavu činí 25 mm.



Obrázek 2.7: Magnetický kontakt MC-38 [26]

Magnetický kontakt je možné připojit šroubovacím konektorem na obrázku 2.8, který je v obvodu zapojen s 470 k $\Omega$  pull-up rezistorem. Tím je např. oproti využití interního pull-up rezistoru mikrokontroléru snížen průtok proudu v případě sepnutí kontaktu. Magnetický kontakt je možné v zapojení nahradit také mechanickým kontaktem. Konektor je připojen k pinu D3 Arduino desky, na kterém je možné vyvolání přerušení mikrokontroléru.



Obrázek 2.8: Schéma zapojení konektoru pro magnetický kontakt

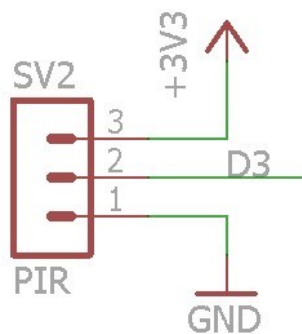
### 2.2.5 Pasivní infračervené čidlo

Použitý modul PIR čidla SB00322A-1 s Fresnelovou čočkou má úhel záběru 100 ° a detekční vzdálenost 2–5 m. Napájecí napětí modulu může být v rozsahu 3,3 V až 15 V a odběr proudu je nižší než 1 mA. Velikost modulu bez čočky je 8 x 10 mm a v případě detekce pohybu je na pinu modulu OUT napětí 3 V. [15]



Obrázek 2.9: Modul PIR čidla SB00322A-1 [15]

PIR čidlo je možné k obvodu připojit konektorem na obrázku 2.10. Jelikož je PIR čidlo připojeno ke stejnému pinu Arduino desky jako konektor pro magnetický kontakt, je na každém sensorovém uzlu možné využít vždy jen jeden z těchto senzorů. Připojení obou senzorů ke stejnému pinu je provedeno z důvodu možnosti vyvolání přerušení mikrokontroléru jen na pinech D2 a D3, přičemž pin D2 je využit pro připojení bezdrátového modulu.



Obrázek 2.10: Schéma zapojení konektoru pro PIR čidlo SB00322A-1

### 2.2.6 Senzor teploty DS18B20

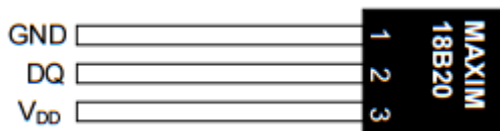
DS18B20 je digitální senzor teploty, který vyrábí firma Dallas Semiconductor. Tento senzor komunikuje prostřednictvím sběrnice 1-Wire navrženou stejnou firmou a pro komunikaci vyžaduje pouze jeden datový vodič. Tato sběrnice je určena pro komunikaci zařízení s nízkou přenosovou rychlostí. Síť, kterou tvoří jedno nebo více 1-Wire slave zařízení a jedno master zařízení se nazývá MicroLan. Pro umožnění komunikace s více senzory na 1-Wire sběrnici má každý senzor DS18B20 v interní paměti ROM uloženo unikátní 64bitové sériové číslo. [42] [43]

Senzor umožňuje měření teplot v rozsahu  $-55\text{ }^{\circ}\text{C}$  až  $+125\text{ }^{\circ}\text{C}$ , přičemž v rozsahu teplot  $-10\text{ }^{\circ}\text{C}$  až  $+85\text{ }^{\circ}\text{C}$  měří s přesností  $\pm 0,5\text{ }^{\circ}\text{C}$ . Senzor umožňuje nastavení 9–12bitového rozlišení teploty, tedy až  $0,0625\text{ }^{\circ}\text{C}$ . Nastavené rozlišení teploty ovlivňuje také dobu převodu teploty do digitální podoby (viz tab. 2.1). Naměřená teplota je kalibrována ve stupních Celsia. [43]

Tabulka 2.1: Vliv nastavení rozlišení teploty na dobu převodu teploty

rozhlišení teploty	max. doba převodu
9 b	93,75 ms
10 b	187,5 ms
11 b	375 ms
12 b	750 ms

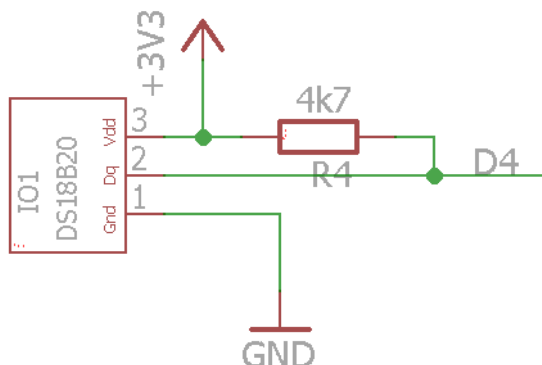
Napájení senzoru DS18B20 je možné napájecím zdrojem stejnosměrného napětí připojeným k pinu  $V_{DD}$  nebo je možné využít režimu „parazitního napájení“, ve kterém je senzor napájen z datového vodiče. Rozsah napájecího napětí je 3–5,5 V. K datovému vodiči (DQ) je navíc potřeba připojit pull-up rezistor. [43]



Obrázek 2.11: Senzor teploty DS18B20 (pouzdro TO-92) [43]

Pro měření senzoru byly použity Arduino knihovny OneWire [27] a DallasTemperature [28].

Zapojení teplotního senzoru DS18B20 uvedené na obrázku 2.12 je provedeno podle katalogového listu výrobce. Napájecí napětí je 3,3 V z napěťového stabilizátoru, pin senzoru Dq je připojen k pinu D4 Arduino desky a pull-up rezistoru R4.



Obrázek 2.12: Schéma zapojení senzoru teploty DS18B20

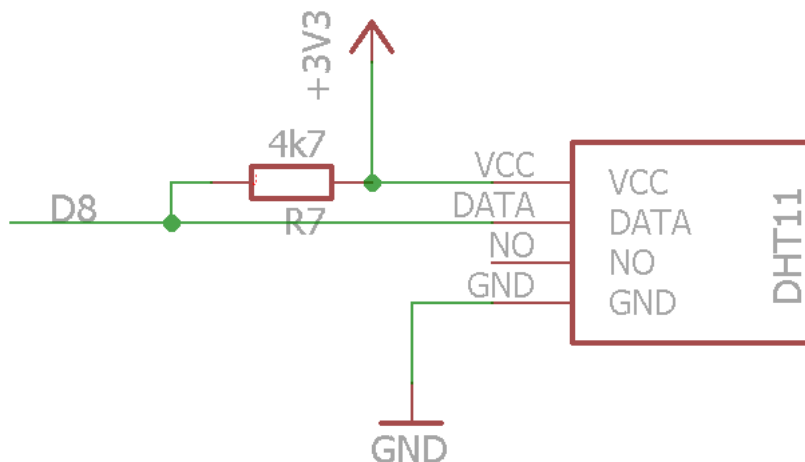
### 2.2.7 Senzor teploty a vlhkosti DHT11

DHT11 je cenově dostupný digitální senzor teploty a vlhkosti. Pro komunikaci s mikrokontrolérem používá stejně jako senzor DS18B20 jeden datový vodič, sběrnice však nejsou kompatibilní a oproti sběrnici 1-Wire např. nepodporuje adresaci.

Senzor umožňuje měření teploty v rozsahu 0 až 50 °C s přesností  $\pm 2$  °C a relativní vlhkosti vzduchu v rozsahu 20 až 90 % s přesností  $\pm 5$  %. Pracovní napětí senzoru je 3,3–5 V a odběr proudu v klidovém režimu 100–150  $\mu$ A. [30]

Senzor je pro nízkou přesnost a rozsah měření teploty vhodný zejména pro měření vlhkosti, je však možné jej zaměnit za přesnější a úspornější senzor DHT22.

Pro měření hodnot ze senzoru byla použita Arduino knihovna DHT [29]. Zapojení senzoru na obrázku 2.13 je provedeno podle katalogového listu výrobce, pin senzoru DATA je připojen k pull-up rezistoru R7 a pinu D8 Arduino desky. Napájecí napětí je 3,3 V z napěťového stabilizátoru připojené k pinu VCC.



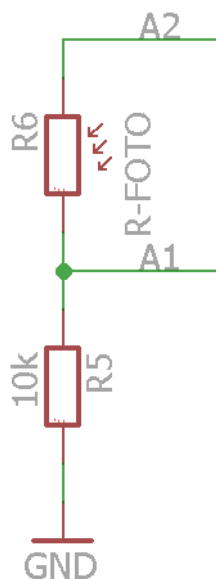
Obrázek 2.13: Schéma zapojení senzoru teploty a vlhkosti DHT11



### 2.2.8 Fotorezistor

Fotorezistor je polovodičová součástka, jejíž odpor klesá se zvyšující se intenzitou dopadajícího světla. V praktickém použití může sloužit např. k detekci setmění a rozednění. V této práci je použit fotorezistor TOKEN PGM5626D jehož odpor při světelném toku 10 lx je 8-20 k $\Omega$  a za tmy 2 M $\Omega$ . [44]

Měření fotorezistoru je v zapojení provedeno napětovým děličem na obrázku 2.14, jehož výstup je připojen k pinu A1 Arduino desky. Pro minimalizaci spotřeby energie je napětový dělič připojen k pinu A2 Arduino desky, kterým je v případě měření přivedeno napájecí napětí.



Obrázek 2.14: Schéma zapojení fotorezistoru

## 2.3 Napájení senzorového uzlu

U senzorového uzlu se předpokládá napájení baterií, je proto kladen důraz na co nejnížší spotřebu energie. Té je možné dosáhnout zejména využíváním úsporných režimů jak u mikrokontroléru desky Arduino, tak u bezdrátového modulu.

Pokud je modul RFM69W uveden do režimu spánku, nemůže přijímat příkazy z ovládacího systému. V případě senzorového uzlu, na kterém je využito funkce spínače či stmívače, není tedy režim spánku bezdrátového modulu využíván a je vhodné pro napájení použít síťový zdroj. Porovnání odběru proudu modulu RFM69W pro různé provozní režimy je uvedeno v tabulce 2.2 [20].

Tabulka 2.2: Odběr proudu modulu RFM69W

režim modulu RFM69W	odběr proudu
režim spánku	0,1 $\mu$ A
příjem dat	16 mA
odesílání dat	16–45 mA

Senzorový uzel, který pouze odesílá data z připojených senzorů ovládacímu systému, může po většinu času setrvávat v režimu spánku a přepínat se do aktivního režimu jen v případě samotného

měření a odesílání dat. Tímto způsobem je dosaženo podstatného snížení spotřeby, jelikož měření je prováděno v intervalu nastaveném pro konkrétní senzor a trvá obvykle zlomek doby provozu v režimu spánku. Režim spánku senzorového uzlu je nastaven po dobu 8 s, v aktivním režimu poté proběhne kontrola, zda je podle nastaveného intervalu měření pro jednotlivé senzory potřeba provést nové měření nebo odeslání dat.

U mikrokontroléru ATmega328P použitým v Arduino Pro Mini je možné pro snížení spotřeby využít některý z šesti režimů spánku (konkrétně idle, ADC noise reduction, power-down, power-save, standby a extended standby). Jednotlivé režimy se mezi sebou liší zejména možnostmi probuzení mikrokontroléru a vypnutím některých jeho částí. [19]

Pro nejvyšší úsporu v klidovém stavu je použit režim power-down se softwarově vypnutou funkcí BOD pro kontrolu napájecího napětí. Pro nastavení úsporného režimu mikrokontroléru je použita knihovna Low-Power [41].

Porovnání naměřeného odběru proudu Arduino Pro Mini v režimech idle a power-down je uvedeno v tabulce 2.3. Naměřený odběr proudu je však při napájení stejnosměrným napětím 3,3 V zejména v režimu power-down znatelně vyšší, než je uvedeno v katalogovém listu výrobce k použitému mikrokontroléru [19]. To je způsobeno některými součástkami osazenými na desce Arduina, zejména pak Power LED, která svým nepřetržitým svícením indikuje provoz Arduina, a stabilizátorem napětí. Tyto komponenty byly tedy z použité Arduino desky odstraněny. Naměřený odběr proudu po této úpravě je uveden v tabulce 2.4.

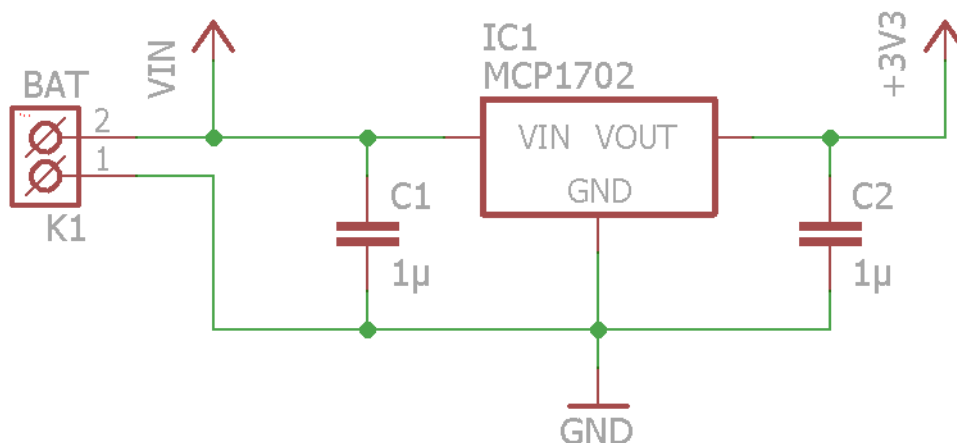
Tabulka 2.3: *Naměřený odběr proudu Arduino Pro Mini ve vybraných provozních režimech*

<b>režim Arduino Pro Mini</b>	<b>odběr proudu</b>
idle	5,04 mA
power-down	1,33 mA

Tabulka 2.4: *Naměřený odběr proudu Arduino Pro Mini po odstranění LED a nap. stabilizátoru*

<b>režim Arduino Pro Mini</b>	<b>odběr proudu</b>
idle	3,67 mA
power-down	0,021 mA

Stabilizátor napětí je v navrženém zapojení nahrazen lineárním stabilizátorem MCP1702T 3302E/TT. Tento stabilizátor se vyznačuje nízkým odběrem proudu v klidovém režimu, vstupním napětím až 13,2 V a výstupním proudem až 250 mA. Zapojení stabilizátoru uvedené na obrázku 2.15 je provedeno podle doporučeného zapojení v katalogovém listu s keramickými kondenzátory C1 a C2 o kapacitě 1  $\mu\text{F}$ . [13]



Obrázek 2.15: Schéma zapojení stabilizátoru napětí MCP1702

### 2.3.1 Baterie

Napájení sensorových uzlů je možné bateriemi nebo externím zdrojem o napětí až 13,2 V. V této práci jsou pro napájení sensorových uzlů použity baterie typu Li-Ion se jmenovitým napětím 3,7 V, které jsou dostupné v různých kapacitách a velikostech. Konkrétně je použito baterií KeepPower typu 14500 o kapacitě 800 mAh a KeepPower typu 18650 o kapacitě 2200 mAh.

### 2.3.2 Měření napětí baterie

Pro nepřesnost interního referenčního napětí A/D převodníku mikrokontroléru ATmega328P bylo jako referenční zvoleno napětí na výstupu stabilizátoru, který také napájí mikrokontrolér. Jelikož měřené napětí baterie je vyšší než referenční napětí A/D převodníku, je pro měření potřeba použít napěťový dělič. Při napájení baterií je potřeba volit hodnoty rezistorů dostatečně velké, aby ji odběr proudu děliče zbytečně nevybíjel. Po dosazení maximálního vstupního napětí stabilizátoru 13,2 V a referenčního napětí A/D převodníku 3,3 V do vzorce 2.3 vypočteme potřebný poměr hodnot rezistorů napěťového děliče.

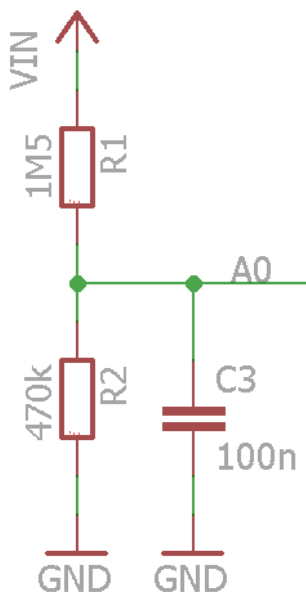
$$U_{výst} = U_{vst} \cdot \frac{R_2}{R_1 + R_2} \quad (2.3)$$

$$3,3 = 13,2 \cdot \frac{R_2}{R_1 + R_2} \quad (2.4)$$

$$3,3R_1 + 3,3R_2 = 13,2R_2 \quad (2.5)$$

$$R_1 = 3R_2 \quad (2.6)$$

Pro zvolenou hodnotu 1,5 MΩ rezistoru R1 podle vzorce 2.6 dopočteme hodnotu rezistoru R2 500 kΩ, v zapojení je tedy použit rezistor R2 s nejbližší nižší běžnou hodnotou 470 kΩ. Jelikož je proud protékající napěťovým děličem velmi nízký, je pro stabilizaci měření v zapojení použit kondenzátor C3 o hodnotě 100 nF. Výstup napěťového děliče je připojen na pin A0 desky Arduino Pro Mini. Schéma zapojení napěťového děliče je na obrázku 2.16.



Obrázek 2.16: Schéma zapojení napěťového děliče

### 2.3.3 Spotřeba senzorového uzlu

Spotřeba jednotlivých senzorových uzlů je závislá především na připojených senzorech, konkrétním nastavení připojených senzorů a bezdrátového modulu a na tom, zda je možné využít režim spánku bezdrátového modulu a mikrokontroléru.

Naměřený odběr proudu z baterie senzorového uzlu s připojeným pohybovým čidlem je v klidovém režimu  $51\ \mu\text{A}$  a při odesílání dat  $21,1\ \text{mA}$ . Pro nastavení odesílání dat, které trvá průměrně  $15\ \text{ms}$  vždy po  $8\ \text{s}$  trvajícím režimu spánku, vychází průměrný odběr proudu  $90,4\ \mu\text{A}$  a teoretická doba provozu na baterii o kapacitě  $800\ \text{mAh}$  tedy dosahuje až  $\sim 368$  dní. Skutečná výdrž baterie však může být ovlivněna např. teplotními podmínkami a samovybitím baterie, které výrobce neuvádí.

## 2.4 Zapojení řídicí jednotky systému

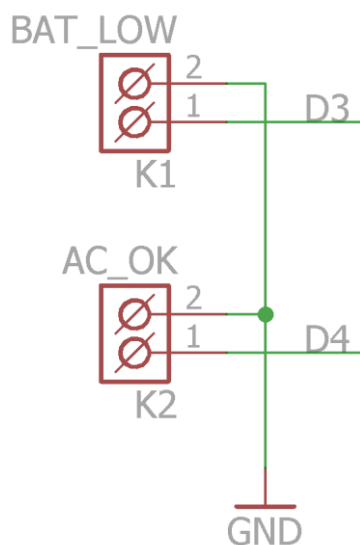
Jako řídicí jednotka systému slouží jednodeskový počítač, ke kterému je sériovým rozhraním připojena deska Arduino Pro Mini. Ta slouží k předávání zpráv přijatých modulem RFM69W ze senzorových uzlů na jednodeskový počítač a naopak.

Napájení Arduino desky a bezdrátového modulu je připojeno k výstupnímu napětí  $3,3\ \text{V}$  napěťového stabilizátoru na jednodeskovém počítači. Pin D4 jednodeskového počítače je propojen s pinem RST Arduino desky pro možnost jejího resetování. K Arduino desce může být také připojena signalizace síťového zdroje popsaného v následující kapitole. Schéma zapojení Arduino brány s jednodeskovým počítačem je uvedeno v příloze B.

## 2.5 Napájení řídicí jednotky systému a IP kamery

Napájení IP kamery a řídicí jednotky je zajištěno spínaným síťovým zdrojem Mean Well PSC 100A-C s funkcí UPS. Zdroj umožňuje nabíjení připojeného akumulátoru o jmenovitém napětí  $12\ \text{V}$ , který v případě výpadku síťového napájení napájí připojená zařízení. Zdroj disponuje signalizací přerušení síťového napájení a nízkého napětí akumulátoru. Tuto signalizaci je možné připojit

ke konektorům Arduino brány podle obrázku 2.17. Na pinech D3 a D4 Arduino desky jsou v případě připojené signalizace použity interní pull-up rezistory.



Obrázek 2.17: Schéma zapojení signalizace síťového zdroje Mean Well

Zdroj obsahuje ochranu proti přetížení, přepětí a úplnému vybití akumulátoru. Další parametry zdroje jsou uvedeny v tabulce 2.5. [16]

Tabulka 2.5: Technické parametry zdroje Mean Well PSC-100A-C

výstupní napětí	13,8 V DC
regulace výstupního napětí	12-15 V
výstupní proud	4,75 A
výstupní proud nabíjení baterie	2,5 A
vstupní napětí	90-264 V AC

Jelikož je pro napájení jednodeskového počítače vyžadováno napětí 5 V, je ke zdroji Mean Well PSC-100A-C připojen měnič stejnosměrného napětí Mean Well SD-15A-5. Parametry tohoto měniče jsou uvedeny v tabulce 2.6. [17]

Tabulka 2.6: Technické parametry DC/DC měniče Mean Well SD-15A-5

výstupní napětí	5 V DC
regulace výstupního napětí	4,75-5,5 V
výstupní proud	3 A
vstupní napětí	9,2-18 V DC

### 2.5.1 Zálohovací akumulátor

Jako zálohovací akumulátor byl ke zdroji Mean Well PSC-100A připojen bezúdržbový olověný akumulátor GLP 26-12. Parametry tohoto akumulátoru jsou uvedeny v tabulce 2.7. [18]

Tabulka 2.7: *Technické parametry akumulátoru GLP 26-12*

jmenovité napětí	12 V
kapacita	26 Ah
nabíjecí napětí	13,5-13,8 V
max. nabíjecí proud	7,2 A
životnost	8-10 let

### 2.5.2 Spotřeba

Spotřeba systému je závislá zejména na počtu napájených kamer, použití jejich IR přísvitu a aktuálním vytížení procesoru jednodeskového počítače. Naměřený odběr proudu IP kamery a stejnosměrného měniče s řídicí jednotkou při napájení zálohovacím akumulátorem je uveden v tabulce 2.8. Odběr proudu stejnosměrného měniče s jednodeskovým počítačem byl měřen po ustálení včetně pevného disku, Arduino brány a GSM modemu. Rozlišení obrazu IP kamery bylo nastaveno na 1920 x 1080 obrazových bodů při snímkové frekvenci 10 fps.

Tabulka 2.8: *Naměřený odběr proudu IP kamery a řídicí jednotky*

<b>zařízení</b>	<b>odběr proudu</b>
IP kamera	0,18 A
IP kamera s IR přísvitem	0,29 A
řídicí jednotka	0,33 A

Podle naměřeného odběru proudu by při výpadku elektřiny mohl být navržený systém s 1 IP kamerou napájen zálohovacím akumulátorem až 50 hodin. Odběr proudu však může být ovlivněn např. nastavením rozlišení obrazu a snímkové frekvence připojené kamery.

## 3 SW implementace a nastavení

V této kapitole je popsána a provedena SW realizace navrženého systému. První částí je instalace operačního systému jednodeskového počítače, kamerového systému ZoneMinder a frameworku ago control se souvisejícími moduly. Je zde také uvedena konfigurace zabezpečeného připojení a propojení použitých systémů. Dále jsou popsány programy navrženého senzorového uzlu, a Arduino brány, modulů frameworku ago control a aplikace pro mobilní telefon s OS Android.

### 3.1 Operační systém jednodeskového počítače

Pro instalaci OS uvedeného v kapitole 1.4.2 byla použita paměťová karta A-Data Premier microSDHC 16 GB Class 10. Instalace je provedena zkopírováním obrazu OS na paměťovou kartu, v OS Linux např. s použitím příkazu `dd if=<ArmbianOS.raw> of=<paměťová_karta>` a spuštěním jednodeskového počítače s připravenou paměťovou kartou. Instalační proces proběhne automaticky a na jeho konci je spuštěn jednoduchý průvodce vytvořením uživatelského účtu. Poté je systém připraven k instalaci dalších částí systému.

Příkazy zadávané v konzoli OS v dalších kapitolách předpokládají přihlášení uživatele jako správce – *root* např. příkazem `su -`.

### 3.2 Instalace kamerového systému ZoneMinder

V repozitářích aktuální 8. stabilní verze Linuxové distribuce není standardně balíček ZoneMinderu k dispozici. Je však dostupný v „backports“ repozitáři, který je možné do OS snadno přidat. Instalaci provedeme podle následujícího návodu.

- Přidáme do systému repozitář „backports“.

```
echo "deb http://http.debian.net/debian jessie-backports main"\
>> /etc/apt/sources.list
```

- Aktualizujeme seznam dostupných balíčků a spustíme samotnou instalaci.

```
apt-get update && apt-get install zoneminder
```

S instalací ZoneMinderu dojde také k instalaci všech potřebných závislostí jako webového serveru Apache, MySQL serveru a knihoven pro zpracování obrazu. Během instalace serveru MySQL má uživatel možnost zvolit heslo pro uživatele *root*.

- Vytvoříme databázi pro ZoneMinder, k čemuž je určen soubor v adresáři ZoneMinderu.

```
mysql -u root -p < /usr/share/zoneminder/db/zm_create.sql
```

- Po instalaci ZoneMinder využívá pro přístup k databázi uživatele *zmuser* s heslem *zmpass*, které je možné změnit v souboru `/etc/zm/zm.conf`. Uživateli je však potřeba povolit k vytvořené databázi přístup.

```
mysql -u root -p -e "grant all on zm.* to 'zmuser'@localhost \
identified by 'zmpass';"
```

- Ve výchozím stavu má ke konfiguračnímu souboru `/etc/zm/zm.conf` přístup jen systémový uživatel `root`. Uživateli (nebo skupině), pod kterým je spuštěn webový server (standardně `www-data`) je potřeba navíc přidat oprávnění ke čtení tohoto souboru.

```
chgrp www-data /etc/zm/zm.conf
```

- V konfiguračním souboru PHP `/etc/php5/apache2/php.ini` vyplníme časové pásmo.

```
sed -i.backup "s/\;date\.timezone =/date\.timezone = \
Europe\Prague/" /etc/php5/apache2/php.ini
```

- Pokud je prvním oddílem připojeného pevného disku souborový systém `ext4`, přidáme jej do souboru `/etc/fstab` a provedeme jeho připojení do adresáře `/var/cache/zoneminder/`.

```
echo -e "/dev/sda1 \t/var/cache/zoneminder\t ext4\t \
defaults\t 1\t 2" >> /etc/fstab && mount -a
```

- Pokud je vlastníkem adresáře `/var/cache/zoneminder/` uživatel `root`, je potřeba jej změnit na uživatele `www-data`.

```
chown -R www-data:www-data /var/cache/zoneminder/
```

- Pro přístup k webovému rozhraní je potřeba povolit připravený konfigurační soubor `/etc/apache2/conf-available/zoneminder.conf` a spouštění skriptů CGI.

```
a2enconf zoneminder && a2enmod cgi
```

- Provedeme restart webového serveru, poté je webové rozhraní ZoneMinderu dostupné na adrese `http://<IP_adresa_serveru>/zm/`

```
service apache2 restart
```



Obrázek 3.1: Hlavní stránka ZoneMinderu po instalaci

### 3.3 Konfigurace kamerového systému ZoneMinder

V systému ZoneMinderu je možné přidávat „monitory“, ty slouží k připojení samotných kamer, přičemž jedna kamera může být připojena k více monitorům. To je vhodné pro případ kdy zvolená kamera podporuje Dual Stream, tedy dva obrazové výstupy v různých kvalitách současně. Těto funkce je možné využít např. pro sledování obrazu na síti s omezenou přenosovou rychlostí nebo pro snížení zatížení procesoru analýzou obrazu tak, že bude analyzován jen méně kvalitní obraz, ale při detekci pohybu bude uložen i záznam s vyšší kvalitou.

Přidání nového monitoru je na hlavní stránce možné tlačítkem *Add New Monitor*. Stránka pro přidání monitoru pak umožňuje podrobné nastavení v záložkách *General*, *Source*, *Timestamp*, *Buffers* a *Misc*.



**Monitor - Monitor-1** Probe Presets

General	Source	Timestamp	Buffers	Misc
Name	Monitor-1			
Server	None ▼			
Source Type	Local ▼			
Function	Monitor ▼			
Enabled	<input checked="" type="checkbox"/>			
Linked Monitors	<div style="border: 1px solid black; height: 40px; position: relative;"> <div style="position: absolute; top: -10px; left: 50%; transform: translateX(-50%);">▲</div> <div style="position: absolute; bottom: -10px; left: 50%; transform: translateX(-50%);">▼</div> </div>			
Analysis FPS	<input type="text"/>			
Maximum FPS	<input type="text"/>			
Alarm Maximum FPS	<input type="text"/>			
Reference Image Blend %ge	6.25% (Indoor) ▼			
AlmRefImageBlendPct	6.25% ▼			
Triggers	None available			

Obrázek 3.2: Stránka ZoneMinderu pro přidání nového monitoru

V záložce *General* je možné nastavit zejména název a funkci monitoru, typ zdroje obrazu, povolení monitoru a připojené monitory, které mají ukládat záznam v případě události na tomto monitoru. Monitor může mít nastaveny následující funkce:

- Monitor – zobrazení aktuálního obrazu ve web. rozhraní, neprobíhá detekce pohybu ani nahrávání
- Modect – záznam obrazu v případě detekce pohybu
- Record – kontinuální záznam obrazu
- Mocord – kontinuální záznam obrazu s detekcí pohybu
- Nodect – funkce pro spouštění záznamu externí událostí

Typ zdroje obrazu lze zvolit *Local*, *Remote*, *File*, *Ffmpeg*, *Libvlc* nebo *cURL* a stejně jako adresu zdroje je třeba jej volit s ohledem na použitý model kamery. Zvolenou IP kameru je možné použít s nastavením typu zdroje obrazu *Ffmpeg* a adresy zdroje s vyplněním uživatele, hesla a IP adresy kamery: `rtsp://<uživatel>:<heslo>@<IP_adresa>/Streaming/Channels/1`. Ukázku nastavení podporovaných modelů kamer je možné nalézt na wiki ZoneMinderu [10].

Záložka *Source* je určena pro nastavení adresy zdroje, eventuálně také např. protokolu, portu, kanálu a formátu. Dále je potřeba nastavit rozlišení obrazu kamery a případně jeho orientaci. V záložce *Timestamp* je možné upravit formát, pozici a velikost časového razítka. Změnu počtu zaznamenaných snímků před a po události je možné provést v záložce *Buffers*. Záložka *Misc* obsahuje doplňující nastavení – např. název událostí a výchozí velikost obrazu kamery ve web. rozhraní.

Webové rozhraní ZoneMinderu po instalaci používá nezabezpečené připojení a nevyžaduje přihlášení uživatele. Přihlašování uživatelů je možné povolit v nastavení *Options – System – OPT\_USE\_AUTH*. Výchozím nastaveným uživatelem je *admin* s heslem *admin*. Změnu hesla, či přidání nových uživatelů je možné provést v menu *Options – Users*. Použití zabezpečeného připojení provedeme následujícím postupem.

- Pro použití zabezpečeného připojení vytvoříme certifikát.

```
mkdir /etc/apache2/ssl/ && openssl req -x509 -nodes -days 365 \
-newkey rsa:2048 -keyout /etc/apache2/ssl/zm.key -out \
/etc/apache2/ssl/zm.crt
```

- Vytvořený certifikát přidáme do souboru default-ssl.conf v adresáři /etc/apache2/sites-available/

```
sed -i "s/SSLCertificateFile.*/SSLCertificateFile\
\/etc\/apache2\/ssl\/zm\.crt/" /etc/apache2/sites-\
available/default-ssl.conf
```

```
sed -i "s/SSLCertificateKeyFile.*/SSLCertificateKeyFile\
\/etc\/apache2\/ssl\/zm\.key/" /etc/apache2/sites-\
available/default-ssl.conf
```

- Povolíme modul SSL web. serveru a zobrazení zabezpečené stránky.

```
a2enmod ssl && a2ensite default-ssl.conf
```

- Restartujeme web. server.

```
service apache2 restart
```

### 3.4 Instalace frameworku ago control

V repozitářích aktuální 8. stabilní verze distribuce Debian není balíček frameworku ago control k dispozici. Framework ago control však poskytuje balíčky ve vlastním repozitáři, který je možné do systému přidat. Instalaci frameworku provedeme následujícím postupem. [12]

- Do OS naimportujeme GPG klíč repozitáře.

```
wget -O - \
http://mirror.at.agocontrol.com/debian/conf/agocontrol.gpg.key
| \ apt-key add -
```

- Přidáme daný repozitář do OS a aktualizujeme seznam dostupných balíčků.

```
echo -e "\ndeb http://mirror.at.agocontrol.com/jessie unstable
\ main" >> /etc/apt/sources.list

apt-get update
```

- Nainstalujeme samotný framework ago control a moduly alert, gammu a zmcam.

```
apt-get install agocontrol agocontrol-alert agocontrol-gammu
agocontrol-zmcam
```

- Pokud již není součástí OS, nainstalujeme službu pro záznam do logu rsyslog.

```
apt-get install rsyslog
```

- Nastavíme záznam logu z frameworku ago control do vyhrazeného souboru a restartujeme službu rsyslog.

```
sed -i.backup \
"s/user.*\/\0\nlocal0\.\.*\t\/var\/log\/agocontrol.log/" \
/etc/rsyslog.conf
```

```
service rsyslog restart
```

- Povolíme automatické spouštění se startem OS a spustíme ago control framework. Web. rozhraní frameworku je dostupné na adrese `http://<IP_adresa_serveru>:8008`.

```
/opt/agocontrol/bin/agoctrl.sh enable
```

```
/opt/agocontrol/bin/agoctrl.sh start
```

### 3.5 Konfigurace zabezpečeného přístupu k frameworku ago control

Přístup k webovému rozhraní frameworku ago control je standardně nezabezpečený ať už ověřením uživatele nebo použitím šifrované komunikace. Pro přihlašování uživatele v rámci lokální sítě je možné použít připravenou aplikaci.

```
/opt/agocontrol/bin/rpc-htpasswd /opt/agocontrol/html/.htpasswd \
agocontrol uzivatel heslo
```

Pro přístup k systému z vnější sítě je vhodné použít reverzní SSL proxy server s ověřováním uživatele. Jelikož kamerový systém ZoneMinder používá webový server Apache, je tento použit také pro konfiguraci proxy serveru. Konfiguraci proxy serveru provedeme v následujících krocích.

- Vytvoříme soubor s uživatelským jménem použitým pro přihlašování.

```
htpasswd -c /etc/apache2/.htpasswd uzivatel
```

- Vytvoříme certifikát pro proxy server.

```
mkdir /etc/apache2/ssl/ && openssl req -x509 -nodes -days 365 \
-newkey rsa:2048 -keyout /etc/apache2/ssl/agocontrol.key -out \
/etc/apache2/ssl/agocontrol.crt
```

- Vytvoříme soubor pro konfiguraci proxy serveru (např. `agocontrol-proxy.conf`) v adresáři `/etc/apache2/sites-available`, případně zkopírujeme a upravíme soubor `default-ssl.conf` v daném adresáři.

- Obsah vytvořeného souboru bez případných komentářů by měl být následující:

```
<IfModule mod_ssl.c>
```

```
<VirtualHost *: [PORT]>
```

```
ProxyPreserveHost On
```

```
ProxyPass / http://localhost:8008/
```

```
ProxyPassReverse / http://localhost:8008/
```

```
ServerName localhost
```

```
<Location />
```

```
AuthType Basic
```

```
        AuthName "ago control proxy"
        AuthUserFile /etc/apache2/.htpasswd
        Require valid-user
    </Location>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
    SSLEngine on
    SSLCertificateFile /etc/apache2/ssl/agocontrol.crt
    SSLCertificateKeyFile /etc/apache2/ssl/agocontrol.key
    <FilesMatch "\.(cgi|shtml|phtml|php)$">
        SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
        SSLOptions +StdEnvVars
    </Directory>
</VirtualHost>
</IfModule>
```

- V sekci VirtualHost si zvolíme port, na kterém bude proxy server naslouchat a přidáme jej do souboru /etc/apache2/ports.conf

```
sed -i.backup "s/Listen\ 443/\0\n\tListen\ PORT/" \
/etc/apache2/ports.conf
```

- Povolíme potřebné moduly web. serveru Apache a provedeme jeho restart.

```
a2enmod proxy ssl proxy_http && service apache2 restart
```

### 3.6 Propojení kamerového systému ZoneMinder s ago control

Pro zobrazení obrazu z kamer v systému ago control a možnost spuštění nahrávání např. při detekci narušení objektu je potřeba nainstalovat balíček agocontrol-zmcam – viz kapitola 3.4. Dále je nutné v menu systému ZoneMinder povolit možnost *Options – System – OPT\_TRIGGERS*.

V souboru /etc/opt/agocontrol/conf.d/zmcam.conf je potřeba specifikovat zvolené nastavení systému ZoneMinder. Ukázka konfiguračního souboru je zobrazena níže.

```
[zmcam]
server=192.168.1.2
webprotocol=https
webport=443
```

```
authtype=hash
username=admin
password=admin
authhashscret="...hash..."
triggerport=6802
monitors=1,2
```

V konfiguračním souboru je možné specifikovat IP adresu serveru, na kterém je systém ZoneMinder nainstalován (nemusí tedy být na stejném serveru jako ago control), způsob ověření a přihlašovací údaje zvoleného uživatele. Jako `triggerport` je potřeba nastavit port, na kterém naslouchá modul `zmtrigger` (standardně 6802). V možnosti `monitors` je třeba určit ID monitorů v systému ZoneMinder.

Po spuštění služby `agozmcam` jsou monitory systému ZoneMinder zobrazeny v menu *Devices* webového rozhraní `ago control`, pro jejich zobrazení na některé z nastavených obrazovek je potřeba vyplnit jejich název a volitelně umístění. Spuštění služby provedeme následujícím příkazem.

```
service agozmcam start
```

### 3.7 Nastavení GSM modemu

GSM modem Huawei E1750 je připojen k USB portu jednodeskového počítače a je používán pro SMS komunikaci s frameworkem `ago control`. Pro jeho použití v modulu pro obsluhu SMS je potřeba zajistit jeho trvalé pojmenování v rámci OS, které správce zařízení pro OS Linux `udev` automaticky nezajišťuje. Požadovaný trvalý název pro vybrané zařízení je možné specifikovat v souboru umístěném v adresáři `/etc/udev/rules.d/`, který je správcem `udev` při připojení zařízení nebo startu systému kontrolován. Soubory umístěné v tomto adresáři musí mít příponu „rules“.

Strukturu souboru a případně další informace o správci zařízení `udev` můžeme zjistit v manuálové stránce zadáním příkazu `man udev`. Vytvoření souboru pravidel provedeme následovně:

- Zjistíme ID výrobce a zařízení modemu, které jsou ve výpisu odděleny znakem „:“

```
lsusb | grep Huawei
```

```
Bus 003 Device 003: ID 12d1:1001 Huawei Technologies Co., Ltd.
E169/E620/E800 HSDPA Modem
```

- Vytvoříme soubor v adresáři `/etc/udev/rules.d/`, zjištěné ID zadáme jako parametry `idVendor` a `idProduct`

```
echo 'SUBSYSTEMS=="usb", ATTRS{idVendor}=="12d1", \
ATTRS{idProduct}=="1001", SYMLINK+="ttyGSM"' > \
/etc/udev/rules.d/60-gsm.rules
```

- Provedeme aplikaci pravidel definovaných ve vytvořeném souboru

```
udevadm trigger
```

### 3.8 Program senzorového uzlu

Program senzorového uzlu slouží k měření hodnot připojených senzorů, obsluhu připojených výstupních zařízení a komunikaci s Arduino deskou připojenou k jednodeskovému počítači, která plní funkci brány. Program podporuje připojení pohybového senzoru nebo magnetického kontaktu, senzorů teploty a vlhkosti, fotorezistoru nebo využití pinů Arduino desky jako univerzální vstup či výstup (např. tlačítko či LED). Dále umožňuje měření napětí baterie, použití PWM výstupu jako stmívače a ovládání bezdrátových spínačů RSL připojeným bezdrátovým modulem RFM85.

Program je navržen s ohledem na možnost připojení různých kombinací senzorů či výstupních zařízení k senzorovému uzlu. Pro překlad vybraných částí zdrojového kódu v závislosti na připojených zařízeních je využito direktiv preprocesoru pro podmíněný překlad.

U senzorového uzlu je předpokládáno využití režimu spánku mikrokontroléru se zastavenými čítači 0-2, ve kterém může setrvat po dobu 8 s. Doba běhu programu je určována počtem průběhů hlavní části programu a pro jednotlivé senzory je tedy možné nastavit interval měření a odesílání dat v násobcích 8 s.

Program pro komunikaci s bezdrátovým modulem využívá knihovnu SPI dostupnou v instalaci Arduino IDE a dále knihovny RFM69 [21], LowPower [41], OneWire [27], DallasTemperature [28], DHT [29] a rc-switch [23].

Zdrojový kód programu je rozdělen do několika částí – inicializace a nastavení, nekonečného cyklu a obsluhy přerušení.

#### 3.8.1 Inicializace a nastavení

Po spuštění programu senzorového uzlu je nejdříve provedena část nastavení a inicializace, která zahrnuje:

- Nastavení pinů mikrokontroléru jako vstupní či výstupní a digitální či analogové pro připojená zařízení a senzory.
- Inicializace použitých knihoven a proměnných pro připojená zařízení.
- Nastavení parametrů bezdrátových modulů.
- Odeslání informačních zpráv na Arduino bránu o spuštění senzorového uzlu a připojených zařízeních.
- Provedení prvotního měření hodnot připojených senzorů a jejich odeslání na Arduino bránu.
- Nastavení obsluhy přerušení pro připojený pohybový senzor či magnetický kontakt

#### 3.8.2 Hlavní část programu

Hlavní částí programu je nekonečný cyklus, který následuje po části inicializace a nastavení. Obsahem této části je obsluha přijatých zpráv z Arduino brány, měření hodnot připojených senzorů a jejich odeslání na Arduino bránu, případně také uvedení bezdrátového modulu a mikrokontroléru do režimu spánku.

Využití režimů spánku mikrokontroléru či bezdrátového modulu je závislé na připojených zařízeních k senzorovému uzlu. Režim spánku bezdrátového modulu není používán, pokud je připojeno některé výstupní zařízení a je potřeba reagovat na příchozí zprávy. Režimu spánku

mikrokontroléru není používán v případě využití PWM výstupu nebo univerzálního digitálního vstupu, např. pro mechanický kontakt, v opačném případě je na konci každého průběhu cyklu uveden do režimu spánku.

V každém průběhu cyklu je prováděna kontrola, zda je podle nastaveného intervalu pro jednotlivé senzory potřeba provést nové měření, případně odeslání hodnot.

### 3.8.2.1 *Obsluha přijatých zpráv*

Obsluha přijatých zpráv je prováděna v nekonečném cyklu, pokud je k sensorovému uzlu připojeno některé výstupní zařízení. V závislosti na typu přijaté zprávy je tato dále zpracována funkcemi pro ovládání univerzálního výstupu, bezdrátových spínačů RSL nebo PWM výstupu. Přijetí zprávy a provedení příkazu je potvrzeno odesláním ACK na Arduino bránu.

### 3.8.2.2 *Měření hodnot připojených senzorů*

Při měření hodnot jednotlivých senzorů je aktuálně naměřená hodnota porovnávána s poslední hodnotou odeslanou na Arduino bránu. Pokud je hodnota rozdílná od poslední odeslané nebo bylo dosaženo intervalu pro odeslání dat, je aktuální hodnota odeslána na Arduino bránu. Pro vybrané senzory je možné povolit vyžadování potvrzení přijetí dat. Typy a obsah zpráv odesílaných sensorovým uzlem jsou popsány v kapitole 3.8.4.

## 3.8.3 *Obsluha přerušení*

Při připojení pohybového senzoru nebo magnetického kontaktu vyvolají změny jejich stavu přerušení mikrokontroléru. Pokud je mikrokontrolér uveden do režimu spánku, vyvolání přerušení jej ukončí a spustí obslužnou funkci. Přerušení může být také vyvoláno při příjmu nové zprávy bezdrátovým modulem.

V obslužné funkci přerušení je provedeno načtení stavu připojeného pohybového senzoru nebo magnetického kontaktu a jeho odeslání na Arduino bránu. Po přijetí potvrzení o doručení zprávy je uložen údaj o čase odeslání stavu senzoru.

## 3.8.4 *Zprávy zasílané ze sensorového uzlu na Arduino bránu*

### MSGTYPE\_NODEPRES (ID 1)

- ACK vyžadováno
- Obsah zprávy: textový řetězec definovaný jako *NODE\_NAME*
- Účel zprávy: informace o spuštění sensorového uzlu a jeho názvu, zapisováno do logu

### MSGTYPE\_SENSPRES (ID 2)

- ACK vyžadováno
- Obsah zprávy: dvourozměrné pole typu byte se záznamy o připojených senzorech - formát záznamu: [*typ zprávy pro zvolený senzor, ID senzoru*]
- Účel zprávy: informace o připojených senzorech po spuštění sensorového uzlu, registrace senzorů v ago control
- Jedna zpráva může obsahovat informace o více senzorech (až do maximální délky zprávy 60 B), případně mohou být senzory rozděleny do více zpráv.

#### **MSGTYPE\_BATT (ID 10)**

- ACK volitelně
- Obsah zprávy: ID senzoru [byte], stav nabití baterie v procentech [byte], napětí baterie [float]
- Účel zprávy: informace o stavu nabití baterie

#### **MSGTYPE\_PIR (ID 20)**

- ACK vyžadováno
- Obsah zprávy: ID senzoru [byte], aktuální stav senzoru [boolean]
- Účel zprávy: informace o stavu pohybového senzoru

#### **MSGTYPE\_DOORSW (ID 21)**

- ACK vyžadováno
- Obsah zprávy: ID senzoru [byte], aktuální stav senzoru [boolean]
- Účel zprávy: informace o stavu magnetického kontaktu

#### **MSGTYPE\_TMP\_DS18 (ID 22)**

- ACK volitelně
- Obsah zprávy: ID senzoru [byte], aktuální teplota [float]
- Účel zprávy: informace o teplotě naměřené senzorem DS18B20

#### **MSGTYPE\_TMP\_DHT (ID 23)**

- ACK volitelně
- Obsah zprávy: ID senzoru teploty [byte], aktuální teplota [float], ID senzoru vlhkosti [byte], aktuální vlhkost v procentech [float]
- Účel zprávy: informace o teplotě a vlhkosti naměřené senzorem DHT11 nebo DHT22

#### **MSGTYPE\_LDR (ID 25)**

- ACK volitelně
- Obsah zprávy: ID senzoru [byte], intenzita osvětlení v procentech [byte]
- Účel zprávy: informace o intenzitě osvětlení na fotorezistoru

#### **MSGTYPE\_INSW (ID 30)**

- ACK volitelně
- Obsah zprávy: ID senzoru [byte], aktuální stav senzoru [boolean]
- Účel zprávy: informace o stavu připojeného tlačítka, případně jiného binárního senzoru (senzor zvuku, kouře, ohně)

### **3.9 Program Arduino brány**

Program Arduino brány slouží k předávání zpráv mezi ago control modulem, dále nazývaným jako ago control klient, popsaným v kapitole 3.10 a senzorovými uzly. Program pro komunikaci s ago control klientem využívá sériové rozhraní, je tedy možné jej využít např. i ve spojení s jinými



jednodeskovými počítači. Program také obsahuje kontrolu signalizace přerušení síťového napájení a nízkého napětí akumulátoru použitého síťového zdroje Mean Well PSC-100A-C.

Program pro komunikaci s bezdrátovým modulem využívá knihovnu SPI dostupnou v instalaci Arduino IDE a knihovnu RFM69 [21].

Zdrojový kód programu je rozdělen jako standardní Arduino programy – do částí inicializace a nastavení a nekonečného cyklu.

### 3.9.1 Inicializace a nastavení

Po spuštění programu Arduino brány je nejdříve provedena část nastavení a inicializace, která zahrnuje:

- Nastavení sériového rozhraní s přenosovou rychlostí 57600 bps.
- Nastavení pinů mikrokontroléru připojených k signalizaci síťového zdroje jako vstupní s povolením interních pull-up rezistorů.
- Nastavení parametrů bezdrátového modulu RFM69W.
- Odeslání informačních zpráv ago control klientovi o spuštění Arduino brány a připojené signalizaci síťového zdroje.

### 3.9.2 Hlavní část programu

Hlavní částí programu je nekonečný cyklus, který následuje po části inicializace a nastavení. Obsahem této části je obsluha přijatých zpráv ze senzorových uzlů, obsluha přijatých dat po sériovém rozhraní a kontrola signalizace síťového zdroje.

#### 3.9.2.1 *Obsluha přijatých zpráv ze senzorových uzlů*

Přijaté zprávy ze senzorových uzlů jsou v závislosti na typu zprávy uvedeném v kapitole 3.8.4 zpracovány funkcemi pro odeslání zprávy po sériovém rozhraní. Pokud je to senzorovým uzlem vyžadováno, je odesláno potvrzení o přijetí zprávy.

#### 3.9.2.2 *Obsluha sériového rozhraní*

Obsluhu přijatých dat ze sériového rozhraní částečně řeší samotný Arduino programovací jazyk. Pokud jsou na sériovém rozhraní dostupná data k přečtení, jsou přidána do bufferu. Tento buffer je poté kontrolován v nekonečném cyklu programu a v případě přečtení znaku nového řádku „\n“ jsou přijatá data označena ke zpracování.

Přijatá data od ago control klienta po sériovém rozhraní jsou kontrolovány na správnost formátu zpráv. V závislosti na jejich typu (viz kapitola 3.9.5) jsou dále příslušnými funkcemi zpracovány a odeslány na senzorový uzel. Po přijetí ACK od senzorového uzlu je provedení požadovaného příkazu potvrzeno zpět ago control klientovi.

#### 3.9.2.3 *Signalizace síťového zdroje*

V každém průběhu cyklu je zkontrolován stav pinů mikrokontroléru připojených k signalizaci přerušení síťového napájení a nízkého napětí akumulátoru použitého síťového zdroje. V případě změny stavu na některém z pinů je tato informace předána ago control klientovi.

### 3.9.3 Zprávy zasílané z Arduino brány na senzorový uzel

#### MSGTYPE\_OUTSW (ID 70)

- ACK vyžadováno
- Obsah zprávy: ID výstupu [byte], požadovaný stav výstupu [boolean]
- Účel zprávy: ovládání bezdrátových modulů RSL, případně jiných výstupních zařízení

#### MSGTYPE\_DIMM (ID 71)

- ACK vyžadováno
- Obsah zprávy: ID PWM výstupu [byte], požadovaná hodnota PWM výstupu v procentech [byte]
- Účel zprávy: ovládání PWM výstupu

### 3.9.4 Zprávy zasílané z Arduino brány ago control klientovi

Všechny zprávy přenášené mezi Arduino bránou a jednodeskovým počítačem po sériovém rozhraní začínají znakem „#” a končí znakem nového řádku „\n”. Hranaté závorky uvedené ve struktuře zpráv dále obsahem samotné komunikace nejsou.

#### MSGTYPE\_NODEPRES (ID 1)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[název odesílatele]\n”

#### MSGTYPE\_SENSPRES (ID 2)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID typu senzoru],[ID senzoru];\n”
- Část zprávy “[ID typu senzoru],[ID senzoru];” se může opakovat

#### MSGTYPE\_BATT (ID 10)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[stav nabití baterie];[napětí baterie]\n”

#### MSGTYPE\_PIR (ID 20)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[stav pohybového senzoru]\n”

#### MSGTYPE\_DOORSW (ID 21)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[stav magnetického kontaktu]\n”

#### MSGTYPE\_TMP\_DS18 (ID 22)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[teplota]\n”

#### MSGTYPE\_TMP\_DHT (ID 23)

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru teploty];[teplota];[ID senzoru vlhkosti];[vlhkost]\n”

#### **MSGTYPE\_LDR (ID 25)**

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[intenzita osvětlení]\n”

#### **MSGTYPE\_INSW (ID 30)**

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID senzoru];[stav senzoru]\n”

#### **MSGTYPE\_OUTSW (ID 70)**

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID spínače];[stav spínače]\n”

#### **MSGTYPE\_DIMM (ID 71)**

- Struktura zprávy: „#[ID brány];[ID odesílatele];[ID zprávy];[ID PWM výstupu];[hodnota PWM výstupu]\n”

### **3.9.5 Zprávy zasílané ago control klientem na Arduino bránu**

#### **MSGTYPE\_OUTSW (ID 70)**

- Struktura zprávy: „#[ID příjemce];[ID zprávy];[ID spínače];[požadovaný stav spínače]\n”

#### **MSGTYPE\_DIMM (ID 71)**

- Struktura zprávy: „#[ID příjemce];[ID zprávy];[ID PWM stmívače];[požadovaná hodnota PWM stmívače]\n”

## **3.10 Modul ago control klient**

Modul ago control klient slouží pro předávání zpráv mezi frameworkem ago control a Arduino bránou a naopak. Modul je napsán v programovacím jazyce Python [32] s využitím standardních knihoven a knihovny agoclient pro komunikaci s ago control, která je součástí instalace ago control frameworku.

Modul je rozdělen do několika tříd, které obsluhují komunikaci s ago control frameworkem a vstup či výstup sériového rozhraní. Po spuštění modulu je navázáno připojení k frameworku ago control. Modul knihovnu agoclient rozšiřuje o funkce pro uchování seznamu připojených zařízení, které po navázání spojení zajistí zobrazení uložených zařízení v ago control např. po startu OS.

Pokud modul detekuje adresářovou strukturu OS armbian pro ovládání GPIO pinů jednodeskového počítače, pokusí se resetovat připojenou Arduino bránu. Poté je provedeno připojení k Arduino bráně podle parametrů nastavených v konfiguračním souboru modulu. Ukázka nastavení konfiguračního souboru `/etc/opt/agocontrol/conf.d/asensors.conf` je uvedena níže.

```
[asensors]
port=/dev/ttyS1
baudrate=57600
```

Po připojení je spuštěna obsluha komunikace s Arduino bránou a příkazů z frameworku ago control.

### 3.10.1 Obsluha příkazů z frameworku ago control

Při odeslání příkazu pro některé z připojených zařízení je knihovnou agoclient volána funkce pro zpracování příkazu. V závislosti na typu příkazu a cílovém zařízení je sestavena zpráva pro odeslání na Arduino bránu a přidána do fronty, formát zpráv je uveden v kapitole 3.9.5. Pokud jsou ve frontě dostupné zprávy k odeslání, pak jsou jednotlivě odesílány v samostatném vláknu modulu. Zpracováváné příkazy pro ovládání zařízení jsou uvedeny v tabulce 3.1.

Tabulka 3.1: *Příkazy pro ovládání zařízení frameworku ago control*

ago control příkaz	popis
setlevel	nastavení hodnoty zařízení v procentech, použito u PWM
on	zapnutí zařízení
off	vypnutí zařízení

### 3.10.2 Příjem zpráv z Arduino brány

Obsluha přijatých zpráv po sériovém rozhraní probíhá v samostatném vlákne modulu. Přijaté zprávy jsou kontrolovány na správnost formátu zpráv a dále zpracovány příslušnými funkcemi. Pokud dojde k odpojení Arduino brány, je v intervalu 10 s prováděn pokus o opětovné připojení. Zpracováváné zprávy a jejich formát jsou uvedeny v kapitole 3.9.4. Přijaté zprávy jsou předány frameworku ago control formou událostí uvedených v tabulce 3.2 s využitím typů zařízení uvedených v tabulce 3.3.

Tabulka 3.2: *Události pro komunikaci s frameworkem ago control*

událost v ago control	popis
event.device.announce	informace o přidání nového zařízení
event.device.batterylevelchanged	informace o stavu nabití baterie
event.security.sensortriggered	informace o stavu binárního senzoru (např. senzor pohybu)
event.environment.temperaturechanged	informace o aktuální teplotě
event.environment.humiditychanged	informace o aktuální vlhkosti
event.environment.brightnesschanged	informace o aktuální intenzitě osvětlení
event.device.statechanged	informace o změně stavu zařízení (spínač, stmívač)

Tabulka 3.3: *Typy zařízení implementované v ago control klientovi*

typ zařízení v ago control	popis
batterysensor	senzor pro měření stavu nabití baterie
binarysensor	binární senzor
brightnesssensor	senzor intenzity osvětlení
dimmer	PWM stmívač
humiditysensor	senzor pro měření vlhkosti
switch	spínač
temperaturesensor	senzor pro měření teploty

### 3.10.3 Nastavení systémové služby

Pro spouštění modulu po startu OS stejně jako frameworku ago control, je potřeba pro modul vytvořit soubor systémové služby umístěný v adresáři `/lib/systemd/system/`. Soubor je možné vytvořit např. zkopírováním a upravením souboru `agorpc.service`. Možná podoba souboru `agoasensors.service` pro modul ago control klient reprezentovaný souborem `/opt/agocontrol/bin/agoasensors.py` je uvedena níže.

```
[Unit]
Description=ago control sensors

[Service]
ExecStart=/opt/agocontrol/bin/agoasensors.py
WorkingDirectory=/opt/agocontrol/bin
StandardOutput=syslog
StandardError=syslog
Restart=always
RestartSec=2
User=root

[Install]
WantedBy=multi-user.target
```

Po vytvoření souboru je potřeba nastavit práva ke spuštění souboru modulu příkazem `chmod u+x /opt/agocontrol/bin/agoasensors.py` a povolit vytvořenou službu v OS příkazem `systemctl enable agoasensors.service`.

## 3.11 Modul pro komunikaci s GSM modemem

Součástí práce je možnost ovládání systému prostřednictvím mobilní sítě GSM. K tomu účelu byl v programovacím jazyce Python vytvořen modul pro framework ago control, který slouží pro komunikaci s GSM modemem Huawei z kapitoly 1.4.5. Modul vychází z již dostupného modulu `agocontrol-gammu` umožňujícím zasílání SMS zpráv z frameworku ago control, jehož instalace byla provedena v kapitole 3.4. Modul využívá Python knihovny `agoclient`, `gammu` a standardní knihovny `time` a `threading`.

Modul umožňuje pomocí SMS zpráv ovládání zařízení dostupných ve frameworku ago control, ovládání modulu zabezpečení, případně odeslání textu SMS zprávy jako událost frameworku ago control. Pro tuto událost, případně konkrétní obsah zprávy, je poté možné definovat libovolné reakce systému. Informace k ovládání zabezpečovacích systémů pomocí SMS zpráv je možné nalézt v [5].

### 3.11.1 Knihovna Gammu

Pro komunikaci s GSM modemem pomocí AT příkazů modul využívá Python knihovnu `gammu`, která vyžaduje vlastní konfigurační soubor `.gammurc` umístěný v adresáři

/etc/opt/agocontrol/gammu. V konfiguračním souboru je potřeba nastavit zejména cestu k zařízení GSM modemu v OS a typ připojení k GSM modemu. Ukázka minimálního potřebného nastavení konfiguračního souboru je uvedena níže.

```
[gammu]
port = /dev/ttyGSM
connection = at
```

Informace o možnostech nastavení a projektu gammu je možné nalézt v manuálové stránce, případně na stránkách projektu. [31]

### 3.11.2 Konfigurační soubor

V konfiguračním souboru modulu je možné nastavit povolená telefonní čísla, požadované heslo pro přijaté zprávy, zasílání potvrzení o provedení definovaných příkazů či heslo k zabezpečovacímu modulu frameworku ago control. Parametry konfiguračního souboru jsou následující:

- **password**: Slouží k nastavení hesla, které je třeba uvést na začátku každého SMS příkazu. Pokud parametr není nastaven, není uvedení hesla v SMS příkazech vyžadováno.
- **valid\_numbers**: Slouží k nastavení telefonních čísel, ze kterých budou SMS příkazy přijímány, více telefonních čísel je možné oddělit čárkou. Pokud není nastaveno žádné, modul zpracovává zprávy z libovolného telefonního čísla.
- **alarm\_pin**: Slouží k nastavení hesla zabezpečovacího modulu frameworku ago control, které může být odlišné od hesla pro samotné SMS příkazy. Při neuvedení hesla není možné zabezpečovací modul SMS příkazy ovládat.
- **alarm\_send\_reply**: Pokud je parametr nastaven na „1“ nebo „True“, je vykonání SMS příkazů ovládajících zabezpečovací modul potvrzeno zasláním SMS odpovědi na telefonní číslo odesílatele SMS příkazu.
- **device\_control**: Pokud je parametr nastaven na „1“ nebo „True“, je možné SMS příkazy ovládat zařízení dostupná ve frameworku ago control. V opačném případě je SMS zpráva předána frameworku ago control formou události.
- **device\_control\_reply**: Pokud je parametr nastaven na „1“ nebo „True“, je vykonání SMS příkazů ovládajících připojená zařízení potvrzeno zasláním SMS odpovědi na telefonní číslo odesílatele SMS příkazu.

Níže je uvedena ukázková konfigurace souboru `agogammu.conf` umístěného v adresáři `/etc/opt/agocontrol/conf.d/`.

```
[agogammu]
password=1234
valid_numbers=+420605123456
alarm_pin=4321
alarm_send_reply=1
device_control=1
device_control_reply=0
```

### 3.11.3 Funkce modulu

Po spuštění modulu je navázáno spojení s frameworkem ago control a GSM modemem a načteny přijaté SMS zprávy z GSM modemu. Kontrola nově přijatých SMS zpráv probíhá v samostatném vlákně modulu a po přečtení nové zprávy je z paměti modemu smazána. V případě zaplnění paměti modemu přijatými SMS zprávami již po navázání komunikace s modemem je z paměti smazána první nalezená zpráva.

Odesílání SMS zpráv např. v reakci na narušení objektu se řídí nastavením událostí ve frameworku ago control, viz příloha D. Zpracování příchozích SMS zpráv je závislé na jejich formátu. Možný formát SMS zpráv zpracovávaných modulem je následující (uvedení nebo neuvedení hesla se řídí nastavením konfiguračního souboru, lomené závorky obsahem zpráv nejsou):

- „<heslo> **ALARM** <stav\_zabezpečení>“: Aktivace některého ze stavu zabezpečení nastavených v zabezpečovacím modulu.
- „<heslo> <název\_zařízení> **on**“: Zapnutí zařízení podle názvu nastaveného ve frameworku ago control.
- „<heslo> <název\_zařízení> **off**“: Vypnutí zařízení podle názvu nastaveného ve frameworku ago control.
- „<heslo> <text>“: Pokud SMS zpráva neodpovídá žádnému předchozímu formátu, je předána formou události frameworku ago control.

## 3.12 Mobilní aplikace pro OS Android

Pro možnost ovládání systému založeného na frameworku ago control prostřednictvím mobilního telefonu s OS Android byla vytvořena aplikace v programovacím jazyce Python [32].

Aplikace kromě standardních Python knihoven využívá také open source Python framework Kivy [33], který slouží k vytvoření multiplatformních grafických aplikací jak pro mobilní OS Android a iOS, tak pro desktopové OS Windows, Linux a OS X. Pro sestavení vytvořených aplikací pro mobilní OS Android poskytuje Kivy framework nástroj Buildozer.

Tvorba grafického rozhraní je možná v kódu Python aplikace nebo ve vlastním jazyce frameworku označovaném jako KV. Grafické prvky Kivy frameworku se označují jako widgety a pro použití ve vybrané aplikaci je možné je jednotlivě importovat. Mimo widgety dostupné v samotném

Kivy frameworku je v aplikaci použit také widget `NavigationDrawer`, který slouží k vytvoření bočního vysouvacího menu aplikace. [34]

Aplikace byla testována na mobilních telefonech s OS Android verze 2.3.7 a 5.0.2. Funkčnost byla ověřena také v desktopovém OS Linux Debian verze 8.

Pro komunikaci aplikace s frameworkem `ago control` je využito JSON-RPC API, které je poskytováno jeho webovým serverem (standardně na portech 8008 a 8009). Aplikace implementuje příkazy uvedené v tabulkách 3.1 a 3.4.

Tabulka 3.4: *Příkazy použité pro komunikaci s frameworkem ago control*

<b>ago control příkaz</b>	<b>popis</b>
inventory	požadavek na získání kompletního výpisu zařízení
subscribe	požadavek na přihlášení odběru událostí z ago control
getevent	požadavek na získání události z ago control
unsubscribe	požadavek na odhlášení odběru událostí z ago control

### 3.12.1 Funkce aplikace

Po spuštění aplikace je zobrazena obrazovka sloužící k nastavení parametrů připojení k ago control. Pro připojení je nutné vyplnit nejméně adresu serveru a port, v případě nastaveného ověřování uživatele pak také uživatelské jméno a heslo. Zadané parametry připojení je možné uložit do souboru v zařízení, ze kterého budou při dalším spuštění aplikace automaticky načteny. Uložení parametrů je možné povolit zatržením checkboxu umístěného nad tlačítkem „Login“. Naopak smazání uložených hodnot je možné po stisknutí tlačítka „Login“, pokud jsou všechna textová pole na obrazovce prázdná.

Aplikace umožňuje pro připojení k ago control využít jak zabezpečeného, tak nezabezpečeného spojení. Pro použití zabezpečeného spojení je potřeba v mobilním telefonu s OS Android importovat certifikát `agocontrol.crt` vytvořený v kapitole 3.5. Postup instalace certifikátu je závislý na verzi OS Android a výrobci zařízení, návod pro zařízení Nexus je možné nalézt na [36].

Pokud nemůže být připojení k ago control provedeno, uživatel je o chybě informován ve spodní části obrazovky. Pokud je pokus o připojení k ago control úspěšný, aplikace po načtení potřebných dat přidá do bočního vysouvacího menu položky pro zobrazení zařízení odpovídající nastaveným obrazovkám v ago control. Podporované typy zařízení jsou stejné jako v případě modulu ago control klient a jsou uvedeny v tabulce 3.3.

Ago control pro zobrazení jednotlivých typů zařízení používá volně dostupné ikony z balíku entypo [37], pro napodobení vzhledu frameworku jsou tyto ikony použity i v mobilní aplikaci. Ukázka obrazovky s nastavenými zařízeními je na obrázku 3.3.





Obrázek 3.3: Zobrazení zařízení v Android aplikaci

Při navázaném spojení se aplikace metodou *getevent* dotazuje na nové události v systému, jakékoliv změny hodnot sledovaných zařízení se tedy v aplikaci automaticky zobrazí. Níže je uvedena ukázka odpovědi na požadavek *getevent* s využitím JSON-RPC API.

```
{  "jsonrpc": "2.0",
  "result": {
    "event": "event.device.statechanged",
    "instance": "asensors",
    "level": 255,
    "unit": "",
    "uuid": "09ab5f06-b8ba-44e1-a0ab-aca17c4dc3cc"
  },
  "id": 2 }
```

## Závěr

Tato diplomová práce se zabývá návrhem a realizací alarmního a ovládacího systému, který je možné ovládat prostřednictvím mobilního telefonu. Ve své první části se zabývá představením alarmních systémů, kamerového systému a hlavních částí navrhovaného systému.

Alarmní systém byl navrhován jako bezdrátový z důvodu minimalizace stavebních zásahů při jeho realizaci. Řídicí jednotka systému je tvořena jednodeskovým počítačem s připojeným GSM modemem a deskou Arduino s bezdrátovým modulem. Arduino deska slouží pro bezdrátovou komunikaci s ostatními prvky systému a je možné ji po jednoduché úpravě zapojení použít také s jinými jednodeskovými počítači.

Zapojením Arduino desky s jednodeskovým počítačem a bezdrátovým modulem se zabývá druhá kapitola. V této části je také popsáno zapojení Arduino desky, jednotlivých senzorů a jejich vlastností ve funkci senzorového uzlu. Dále se věnuje také napájení jednotlivých částí systému.

Ve třetí kapitole je provedena instalace, konfigurace a propojení frameworku ago control a kamerového systému ZoneMinder včetně nastavení zabezpečeného připojení. Následuje SW implementace Arduino desek v rolích brány a senzorového uzlu a ago control modulu pro komunikaci s Arduino deskou v roli brány.

Navržený systém je možné ovládat SMS zprávami jejichž formát je uveden v kapitole 3.11.3. Funkci modulu pro ovládání systému pomocí SMS zpráv je možné ovlivnit např. nastavením hesla či povolených telefonních čísel v konfiguračním souboru.

Poslední část práce se zabývá implementací aplikace pro OS Android, která byla vytvořena v programovacím jazyku Python s využitím frameworku Kivy. Aplikace umožňuje ovládání zařízení a sledování hodnot senzorů navrženého systému. Funkčnost aplikace byla otestována také v desktopovém OS Linux Debian verze 8.

Realizovaný alarmní a ovládací systém je možné dále jednoduše rozšířit např. o další prvky bezdrátového systému RSL, případně systémy jiných výrobců. Pro připojení dalších senzorů či zařízení k desce Arduino je možné systém jednoduše rozšířit o další typy zpráv. V budoucnu může realizovaný systém ovládat také AV techniku, případně s použitím RFID čipů fungovat jako přístupový.

## Použitá literatura

- [1] UHLÁŘ, Jan. *Technická ochrana objektů*. Praha: Policejní akademie české republiky, 2005. ISBN 80-7251-189-0.
- [2] KŘEČEK, Stanislav. *Příručka zabezpečovací techniky*. Vyd. 2. [S.l.: s.n.], 2003. ISBN 80-902938-2-4.
- [3] FENNELLY, Lawrence J. *Effective physical security*. 3rd ed. Boston: Elsevier Butterworth Heinemann, c2004. ISBN 0750677678.
- [4] BAKER, Paul R. a Daniel J. BENNY. *The complete guide to physical security*. Boca Raton: CRC Press, 2013. ISBN 9781420099638.
- [5] KREJČÍŘÍK, Alexandr. *SMS: střežení a ovládání objektů pomocí mobilu a SMS : GSM pagery a alarmy : princip použití, návody, příklady*. Praha: BEN - technická literatura, 2004. ISBN 80-7300-082-2.
- [6] LinkSprite pcDuino3 Nano – LinkSprite. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.linksprite.com/linksprite-pcduino3-nano/>
- [7] pcDuino3 nano – armbian.[online]. [cit. 2016-06-25]. Dostupné z: <http://www.armbian.com/pcduino3nano/>
- [8] Arduino – Home. [online]. [cit. 2016-06-25]. Dostupné z: <https://www.arduino.cc>
- [9] ZoneMinder – Home. [online]. [cit. 2016-06-25]. Dostupné z: <https://zoneminder.com/>
- [10] ZoneMinder Wiki – Wiki – Hardware Compatibility List. [online]. [cit. 2016-06-25]. Dostupné z: [https://wiki.zoneminder.com/Hardware\\_Compatibility\\_List](https://wiki.zoneminder.com/Hardware_Compatibility_List)
- [11] DS-2CD2032-I . [online]. [cit. 2016-06-25]. Dostupné z: <http://overseas.hikvision.com//uploadfile/image/20150911103422996.PDF>
- [12] ago control wiki. [online]. [cit. 2016-06-25]. Dostupné z: [https://wiki.agocontrol.com/index.php/Main\\_Page](https://wiki.agocontrol.com/index.php/Main_Page)
- [13] 250 mA Low Quiescent Current LDO Regulator. [online]. [cit. 2016-06-25]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/22008E.pdf>
- [14] Arduino – ArduinoBoardProMini. [online]. [cit. 2016-06-25]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardProMini>
- [15] SB00322A-1. [online]. [cit. 2016-06-25]. Dostupné z: <http://sa.tipa.eu/datasheet/08831013-datasheet-en.pdf>
- [16] 100W Single Output with Battery Charger(UPS Function). [online]. [cit. 2016-06-25]. Dostupné z: <http://www.czech-meanwell.cz/fotky10575/fotov/PSC-100-spec.pdf>
- [17] 15W Single Output DC-DC Converter. [online]. [cit. 2016-06-25]. Dostupné z: [http://www.meanwell.com/mw\\_search/sd-15/sd-15-spec.pdf](http://www.meanwell.com/mw_search/sd-15/sd-15-spec.pdf)
- [18] GLP 26-12. [online]. [cit. 2016-06-25]. Dostupné z: <http://glpower.eu/wp-content/uploads/2014/03/GLP-26-12.pdf>

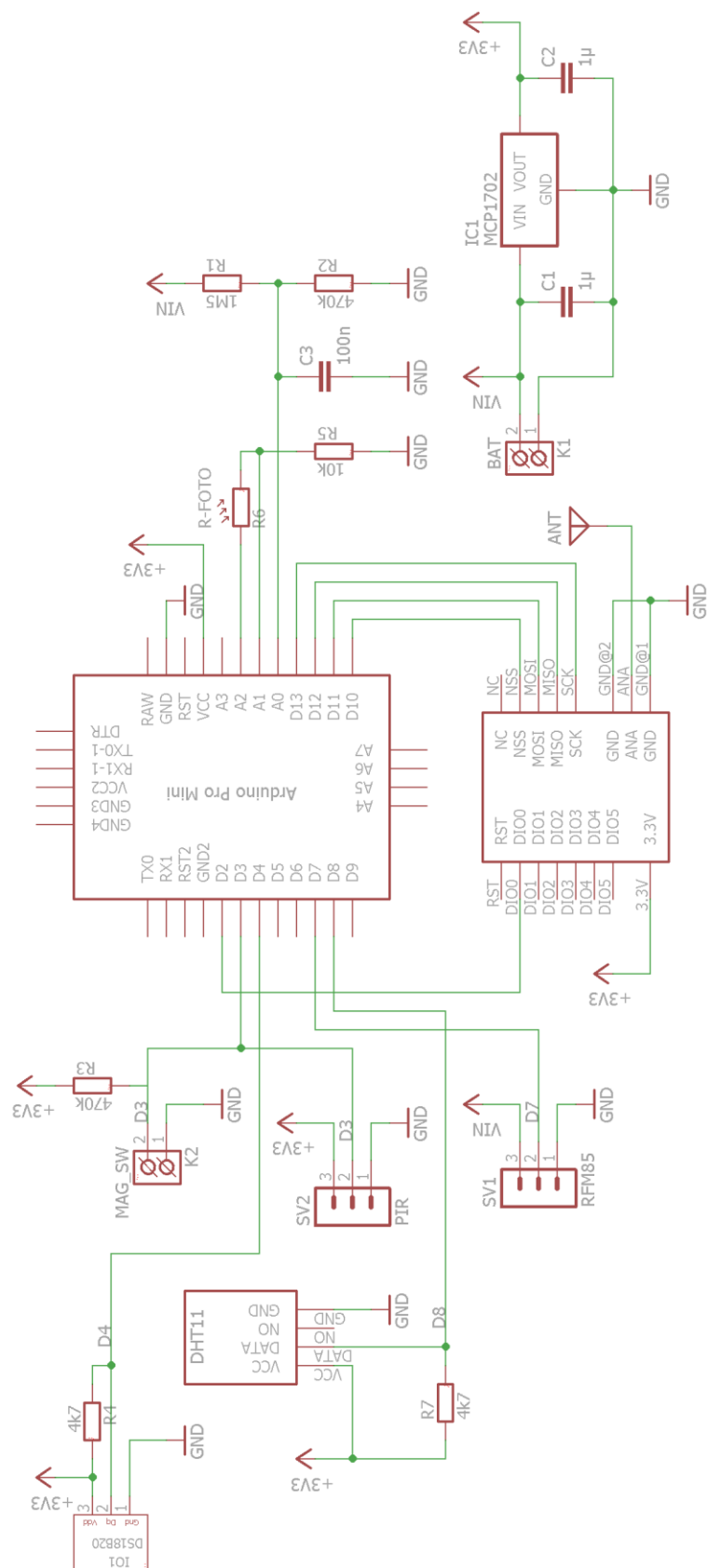
- [19] ATmega48A/PA/88A/PA/168A/PA/328/P. [online]. [cit. 2016-06-25]. Dostupné z: [http://www.atmel.com/images/atmel-8271-8-bit-avr-microcontroller-atmega48a-48pa-88a-88pa-168a-168pa-328-328p\\_datasheet\\_complete.pdf](http://www.atmel.com/images/atmel-8271-8-bit-avr-microcontroller-atmega48a-48pa-88a-88pa-168a-168pa-328-328p_datasheet_complete.pdf)
- [20] RFM69W ISM TRANSCEIVER MODULE V 1 . 3. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.hoperf.com/upload/rf/RFM69W-V1.3.pdf>
- [21] GitHub – LowPowerLab/RFM69: RFM69 library for RFM69W, RFM69HW, RFM69CW, RFM69HCW (semtech SX1231, SX1231H). [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/LowPowerLab/RFM69>
- [22] ASK/OOK TRANSMITTER MODULE RFM85. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.hoperf.de/upload/rf/RFM85.pdf>
- [23] GitHub – sui77/rc-switch: Arduino lib to operate 433/315Mhz devices like power outlet sockets. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/sui77/rc-switch>
- [24] WD Scorpio® Blue™ Mobile Hard Drives. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.wdc.com/wdproducts/library/SpecSheet/ENG/2879-701278.pdf>
- [25] Wireless flush-mounted switch “RSLU”. [online]. [cit. 2016-06-25]. Dostupné z: [http://www.produktinfo.conrad.com/datenblaetter/625000-649999/640302-an-01-ml-FUNK\\_EINBAUSCHALTER\\_RSLU\\_de\\_en\\_fr\\_nl.pdf](http://www.produktinfo.conrad.com/datenblaetter/625000-649999/640302-an-01-ml-FUNK_EINBAUSCHALTER_RSLU_de_en_fr_nl.pdf)
- [26] MC-38 Wired Door Window Sensor Magnetic Switch Home Alarm System. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.oddwires.com/mc-38-wired-door-window-sensor-magnetic-switch-home-alarm-system/>
- [27] GitHub – PaulStoffregen/OneWire: Library for Dallas/Maxim 1-Wire Chip. [s. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/PaulStoffregen/OneWire>
- [28] GitHub – milesburton/Arduino-Temperature-Control-Library: Arduino plug and go library for the Maxim (previously Dallas) DS18B20 (and similar) temperature ICs. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/milesburton/Arduino-Temperature-Control-Library>
- [29] GitHub – markruys/arduino-DHT: Efficient DHT library for Arduino. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/markruys/arduino-DHT>
- [30] DHT11 Humidity & Temperature Sensor. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.micro4you.com/files/sensor/DHT11.pdf>
- [31] Gammu. [online]. [cit. 2016-06-25]. Dostupné z: <http://wammu.eu/gammu/>
- [32] Welcome to Python.org. [online]. [cit. 2016-06-25]. Dostupné z: <https://www.python.org/>
- [33] Kivy framework — Kivy 1.9.2-dev0 documentation. . [online]. [cit. 2016-06-25]. Dostupné z: <https://kivy.org/docs/api-kivy.html>
- [34] GitHub – kivy-garden/garden.navigationdrawer: Kivy widget to duplicate popular android hidden panel functionality. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/kivy-garden/garden.navigationdrawer>
- [35] RPC – ago control wiki. [online]. [cit. 2016-06-25]. Dostupné z: <https://wiki.agocontrol.com/index.php/RPC>

- [36] Práce s certifikáty – Náповěda Nexus. [online]. [cit. 2016-06-25]. Dostupné z: <https://support.google.com/nexus/answer/2844832?hl=cs>
- [37] Entypo. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.entypo.com/>
- [38] GitHub – adafruit/Adafruit-Eagle-Library. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/adafruit/Adafruit-Eagle-Library>
- [39] RFM69 module Eagle library – PanSat. [online]. [cit. 2016-06-25]. Dostupné z: <http://pansat.panska.cz/rfm69-eagle-library/>
- [40] PaJa – Eagle. [online]. [cit. 2016-06-25]. Dostupné z: <http://www.paja-trb.cz/eagle/index.html>
- [41] GitHub – rocketscream/Low-Power: Low Power Library for Arduino. [online]. [cit. 2016-06-25]. Dostupné z: <https://github.com/rocketscream/Low-Power>
- [42] Overview of 1-Wire Technology and Its Use. [online]. [cit. 2016-06-25]. Dostupné z: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/1796>
- [43] DS18B20 Programmable Resolution 1-Wire® Digital Thermometer. [online]. [cit. 2016-06-25]. Dostupné z: <http://cdn.sparkfun.com/datasheets/Sensors/Temp/DS18B20.pdf>
- [44] CDS Light-Dependent Photoresistors. [online]. [cit. 2016-06-25]. Dostupné z: <http://token.com.tw/pdf/resistor/cds-resistor-pgm.pdf>

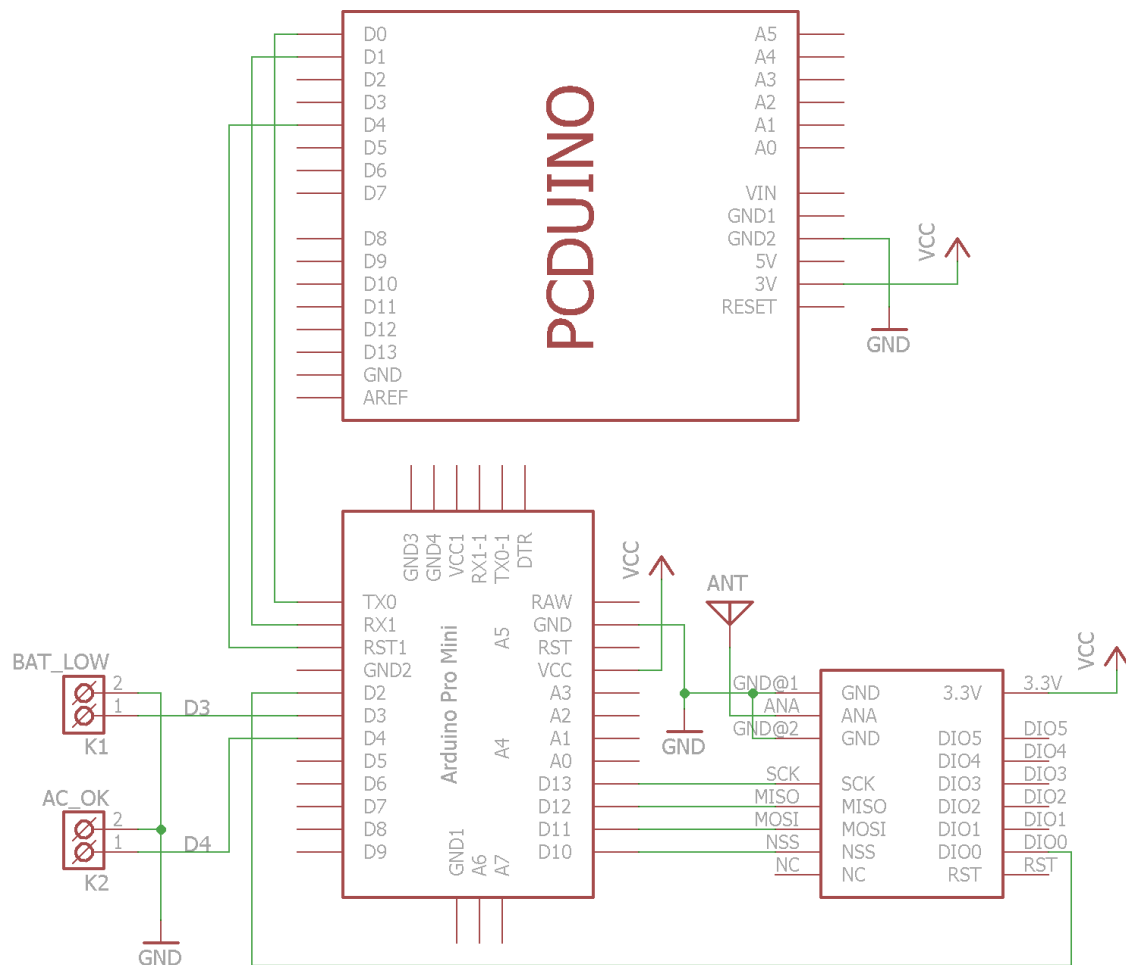
## Seznam příloh

Příloha A:	Schéma zapojení senzorového uzlu .....	I
Příloha B:	Schéma zapojení Arduino brány s jednodeskovým počítačem pcDuino3 Nano ..	II
Příloha C:	Prototyp řídicí jednotky a senzorového uzlu .....	III
Příloha D:	Framework ago control .....	IV

Příloha A: Schéma zapojení senzorového uzlu



Příloha B: *Schéma zapojení Arduino brány s jednodeskovým počítačem pcDuino3 Nano*

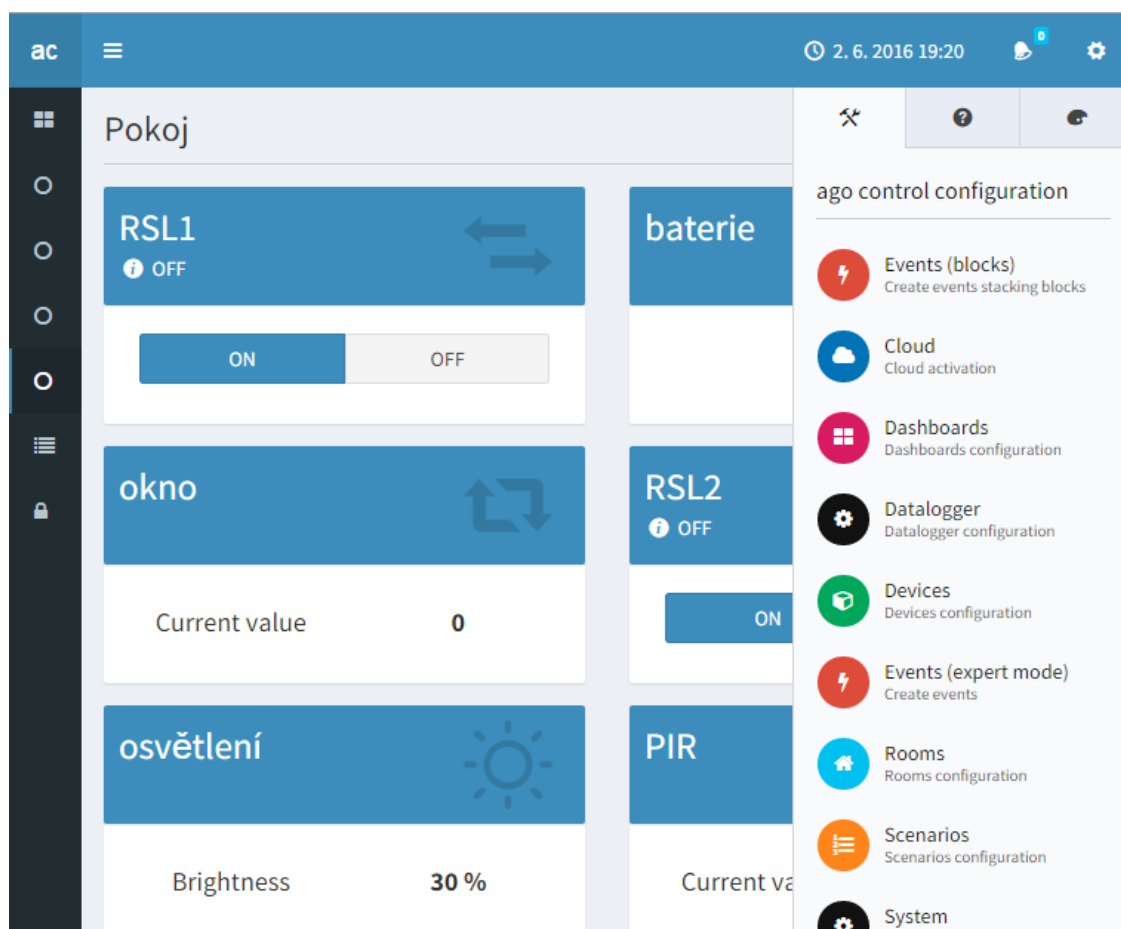




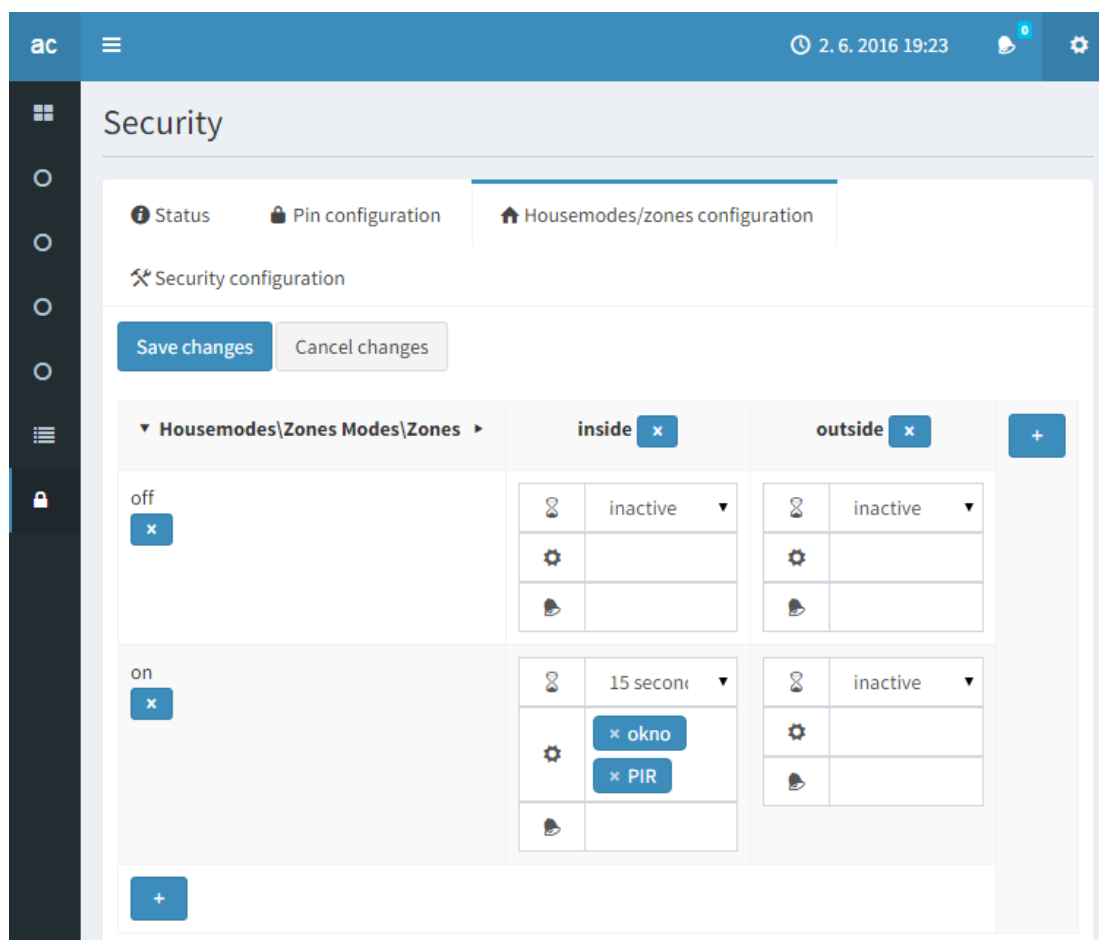
---

Příloha C:      *Prototyp řídicí jednotky a senzorového uzlu*





Obrázek 1: Hlavní obrazovka web. rozhraní frameworku



Obrázek 2: Obrazovka nastavení security modulu

---

Edit event

**Criteria**

intruder alarm ▼

```
{
  and or Add criteria Add nesting delete < >
  {
    event ▼ zone ▼ = ▼
    inside delete
  }
}
```

**Action**

GSM modem ▼

send SMS ▼

**message:**

Pohyb uvnitř domu

**recipient:**

123456789

Cancel

Save

*Obrázek 3: Nastavení události při narušení*